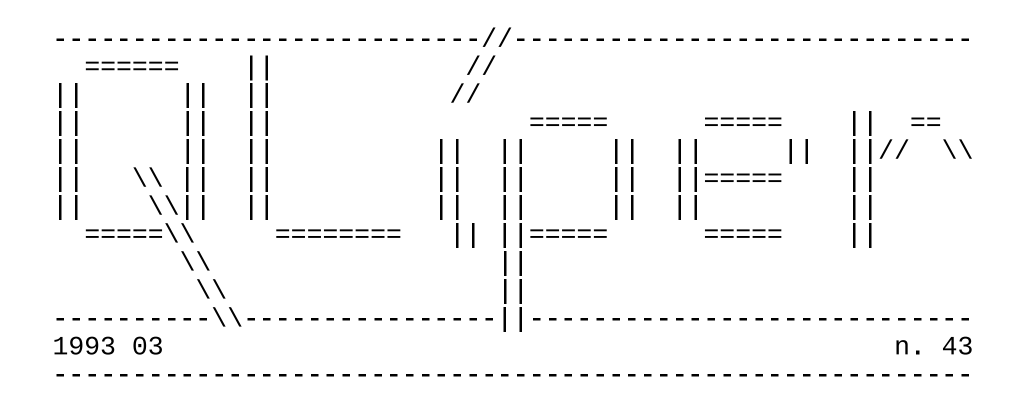


QLíper
Acacias 44 (Monteclaro)
ES-28223 Pozuelo de Alarcón
Tel. +34-(9)1-7153873

NOTAS DESDE LA COMPUGIDA REDACCION

Poco más abajo leeréis una carta de Miguel Estarellas cargada de razón, que me ha hecho pensar los últimos días, más de lo que ya me he comido el seso desde que empecé a ocuparme de la revista. No digo en qué día de abril estoy escribiendo esto, para que no se me caiga la cara de vergüenza, pues se supone que este QLíper es el de marzo...

No es justo que todo este retraso (que ya lleva varios números) se deba a una sola persona, o sea a mí. Quien se comprometa a hacer el montaje de la revista, debe ser capaz de sacarla más o menos a tiempo.

¿Será que en casa del herrero...? ¿O será que al leer los comentarios en IQLR sobre la QXL, lo he dado todo por perdido, como si ya no mereciera más la pena

darle a las teclas de mi viejo QL, hasta poder ahorrar para adquirir esa maravilla? Tendría entonces en una sola máquina el emulador de ZX SPECTRUM para PC (otra maravilla alucinante, adquirida hace poco, que corre los programas de Spectrum 5 veces más rápido que un Spectrum real, en el 486 a 50 MHz) y la QXL. Se acabaron lo cables, los bloqueos, los cuelgues, las esperas, las incompatibilidades... Spectrum, QDOS, DOS y Windows en un solo equipo.

¿O será que, completamente alucinado por mi última adquisición, el gestor de base de datos Microsoft Access para Windows 1.0, se me hace muy cuesta arriba decidirme a sufrir con la rudimentariedad de Archive? Algo que haces en 20 segundos con Access de forma intuitiva sin pulsar una sola tecla, sin más que hacer bailar un poco al ratón, precisa en Archive ponerte a escribir un puñado de aburridos, torpes y lentos procedimientos. Mejor dejarlo, me digo a mí mismo, para cuando me apetezca viajar 8 años hacia atrás en el túnel del tiempo.

¿O será porque, en suma, sé que de llevar la composición de la revista con WordPerfect y la base de datos con Access (o incluso con Clipper), todo se haría más rápidamente y mejor?

Así me siento, lo siento, amigos... Sé que son pensamientos tentadores y herejes, ponedme vosotros mismos la penitencia, pero sed indulgentes. Creo que podréis enterderme.

Pero la causa principal de todo este retraso es más mundana, si cabe: las normales ocupaciones diarias, que en la mayoría de los casos se apoderan también de los fines de semana.

En cuanto al formato, me parece que siempre damos vueltas alrededor del mismo sitio. Efectivamente, el sistema usado en los últimos números hace que cualquiera pueda imprimir la revista sin problemas, incluso si no dispone de un editor en donde cargarla, incluso sin tener memoria suficiente para leerla entera del disco. ¿Pero es eso tan importante? Somos pocos, todos tenemos discos, memoria suficiente, y podemos apañárnoslas solitos con cualquier formato de texto.

Como habréis visto, en este número he pasado de formatear a la derecha (para evitar los espacios adicionales, incómodos si se importa el texto en otro programa), y de meter número de página (terrible labor, incluso con ayuda de The Editor). Además, he normalizado los encabezamientos de los artículos, de modo un pequeño programa en The Editor puede construir solito el índice:

```
rp(f.==.;sm;p;bs;be;gb;bi;cm;n)
```

A propósito: el club escocés es bastante más grande que QLíper, pero sólo edita una página impresa como boletín; nosotros somos catorce y montamos todo este tinglado.

Se admiten todo tipo de ideas y sugerencias:

-Si alguien quiere hacerse cargo de montar la revista, yo estoy dispuesto a plantearme el relevo. No tanto porque no quiera seguir, que sí me gustaría, sino porque, como habéis visto, me es muy difícil ahora, por motivos personales, garantizar la puntualidad, aunque lo intentaré. No obstante, del mismo modo que no tenemos que renunciar a la puntualidad que teníamos antes, tampoco deberíamos renunciar al cuidado en la forma y el contenido que hay ahora. Quien se comprometa a cargar este tinglado sobre sus hombros debe responder de los dos aspectos. Por mi experiencia os diré que llevar todo organizada y cuidadosamente lleva muchas más horas de las que a simple vista parece... no es por asustar.

-Otra solución (a medio plazo inevitable, creo yo) es dejarnos de tanta infraestructura y tanto rollo, para cuatro gatos que somos, y montar una pequeña BBS en que cualquiera pudiera intercambiar material. Bien organizado, no tiene por qué resultar más caro. De hecho cada uno paga su teléfono, así que no habría ni cuota ni nada de nada. No hace falta una BBS normal, sino un simple medio para dejar colaboraciones de todo tipo, y cargar las recibidas hasta el momento. Un tablón de anuncios adicional, y poco más. No obstante, aunque un módem es fácil de adquirir, la mayoría no lo tiene, ni le interesa tenerlo.

-Otra posibilidad: editar la revista en papel, a modo de hoja informativa. Acabaríamos así con muchos problemas, pero también con casi todas las ventajas,

así que mejor olvidarlo.

-Otra (la más interesante a corto plazo): hacer como QITALY, o sea componer cada revista en disco en base a colaboraciones heterogéneas, y usar un programa escrito a propósito para verlas tanto en pantalla como en impresora. De hecho sería fácil hacer un programa mejor que el de ellos (que tiene cosas muy mejorables). De este modo cada cual puede enviar las colaboraciones en formato de Quill o de texto, y no hay que recomponer nada para incluirlas en la revista. Además, cada cual puede ver o imprimir sólo aquellas colaboraciones que le interesen. Y el programa gestor se puede ir perfeccionando poco a poco con más y mejores opciones, para hacerlo más flexible e inteligente. Para que os hagáis una idea exacta, os envío a todos una copia del último número de QITALY.

Bueno, espero que entre todos busquemos una solución, porque merece la pena seguir.

Marcos, abril de 1993 (...si la dicha es buena)

CARTA ABIERTA A QLIPER

¿Es posible que todos estemos de acuerdo con el actual formato de QLíper? Un torpedo más potente: ¿Estamos contentos con el funcionamiento del club?

Creo que sería muy bueno someter a debate todos y cada uno de los aspectos de nuestra organización y funcionamiento. Esto es, evidentemente, muy fácil de decir. Menos fácil es aportar ideas y, sobre todo, llevarlas a la práctica, aunque sólo sea por el cambio que pueda significar.

Nuestra revista es en gran medida nuestro club. Y no hay revista si no colaboramos todos con mayor asiduidad. Hagamos un esfuerzo en este sentido. Y en definitiva tampoco es tanto; todos tenemos algún tema que dominamos y del cual podemos exponer algún aspecto, consultas que realizar, comentarios a artículos anteriores, preguntas que nos gustaría hacer. Por lo tanto el problema es únicamente decidirse a utilizar nuestro editor preferido.

En cuanto a la composición de la revista (con el permiso del Sr. Editor) tengo algún comentario que realizar. Entiendo que el actual formato facilita el paso de la totalidad de QLíper a la impresora, pero tambien obstaculiza la impresión de un artículo aislado.

Quizás fuera positivo crear unas secciones temáticas que, siendo incluso sugerentes para las colaboraciones, den al mismo tiempo más estructura al conjunto de la revista. Posteriormente se podría estudiar la conveniencia de su edición como capítulos separados. ¿Ideas? Algunos "capítulos" ya existen: editorial, buzón. Otros podrían ser software, hardware (estos dos palabros en brindis al Sr. Editor), lenguajes (en este momento merece capítulo aparte el C), mercadillo del QL, etc.

Y no quiero olvidar las reuniones. Creo que son importantes y deben promoverse: ¿qué tal si se fijan de antemano unas fechas anuales? La programación a largo plazo puede facilitar la asistencia de nuestros compañeros más alejados. Desgraciadamente, por el número de socios, creo que sólo pueden tener lugar en Madrid, en cuyo caso el lugar tiene menos importancia y puede decidirse con posterioridad.

Es evidente que hay muchos más aspectos de interés que yo no he tocado en estas notas pero que irán apareciendo en ese debate que sugiero.

26 Enero 1993 Miguel Estarellas (91) 5 73 07 01

REARMANDO EL Z88

Por cierto, ¿Habéis leído el artículo de Marcos "PC 486: dónde están los 8 Megas, matarilerilerile...!?". Yo por mi parte, tuve que leerlo dos veces, pues la primera vez no podía contener la risa. Mi primer ordenador con memoria paginada era un Spectrum 128K de INVES. El manual dejaba mucho que desear

comparado con mi viejo Spectrum 48K, y no explicaba nada de cómo usar los 64 Kbytes extra a excepción de usarlos como RAM-DISK en BASIC. Menos mal que mi SPECTRUM 128K me salió defectuoso (se colgaba a los 15 minutos) y pude cambiarlo por otra cosa en el Corte Inglés. En el caso del Z88, para poder escribir mi Z88 FORTH tuve que comprar el manual del sistema operativo OZ. Allí se explica cómo utilizar los 4 Mbytes que puede direccionar el sistema operativo OZ y dónde están, pero trabajar con segmentos de 16 Kbytes (un Z80 solamente puede direccionar 64 Kbytes a la vez) resulta muy complicado para alguien que ha estado acostumbrado toda su vida a programar bajo memoria lineal.

Me he comprado como regalo de Reyes una TV color de bolsillo marca CASIO para matar un poco mejor el tiempo en mi trabajo, pero solamente he conseguido aburrirme aún más. La autonomía de unas 4 pilas alcalinas LR6 en esa TV es de alrededor de 3 horas. Unas pilas R6 o de esas que venden por alrededor de 100 pesetas el paquete de 4 solamente duran 20-30 minutos. Para utilizar la TV en casa o en la oficina me compré un adaptador AC/DC universal (1.5 a 12 voltios, cambio de polaridad y 6 tipos de clavijas) por 1290 pesetas.

Y me diréis, ¿Por qué tanto rollo de TV?. Bien, podía habérmelo ahorrado, pues forma parte del principio de la historia, pero atentos que la segunda parte es muy interesante.

En la tienda donde me vendieron la TV, me ofrecieron un cargador de pilas R6 de 1.2 voltios por 680 pesetas y un paquete de 4 pilas recargables R6 modelo ÑIKKO NICA de Imcadisa (1,2V 500mAh R-6) por 980 pesetas. Al principio no me atreví a comprar esa opción que no estaba tan mal de precio, pero después de gastar varios paquetes de pilas alcalinas LR6 (con lo caras que están), decidí probar la opción pilas recargables. Sorpresa: la autonomía es una hora y 15 minutos en utilización continua, pero se alcanza la hora y media, o más, en utilización discontinua.

Mi Z88 hace casi un año que no lo estoy usando. Los motivos principales eran dos:

1) Cuando compré mi Z88 (español) solamente tenía un QL ampliado y uno de repuesto sin ampliar. Hoy tengo 3 QLs superampliados en diferentes lugares, y solamente necesito el Z88 en muy raros casos (p.e: en la playa, terraza, calle y campo).

2) Probablemente este sea el motivo por el cual el Z88 se quedó paralizado. El condenado gasta pilas de tal forma que mantenerlo durante dos meses (¡OJO!, utilizándolo solamente una hora diaria y con un cartucho 128 K RAM) puede costar en pilas ALCALINAS tanto como la factura bi-mensual de SEVILLANA ELECTRICIDAD de un apartamento de 100 metros cuadrados.

Advertencia: la última generación de cartuchos de 1 Mbyte de RAM gastan casi igual que la primera generación de cartuchos 128 K RAM. Esto no he podido probarlo, pero eso se dice. Todas formas, si no fuese así, un Z88 con 1 Mbyte de RAM solamente podría estar encendido unos minutos.

4 pilas recargables tipo R6 1,2V 500mAh dan la mitad de autonomía que 4 pilas LR6 de 1,5V. Una caja externa con clavija fuente alimentación externa conteniendo 5 pilas recargables ofrece la misma autonomía que 4 pilas ALCALINAS LR6, pero hay que buscarse la caja (una caja de 6 pilas. Uno de los espacios con un hilo de cobre soldado a los extremos), la clavija, y resulta un poco incómodo y feo.

En resumen, hoy en día las pilas recargables son baratas, y lo más importante, protegen el medio ambiente (un Z88 con cartucho 128K RAM trabajando una hora diaria acumula una basura de 60 pilas ALCALINAS cada 2 meses. Una TV de bolsillo funcionando 2 horas diarias acumula una basura de 120 pilas ALCALINAS cada 2 meses. Las videoconsolas portátiles... mejor no calcularlo, porque se nos van a echar los ecologistas encima).

Por si os interesa. Existen cargadores de pilas R6 o 'AA' solares. RANGER COMPUTER los vende por 102.23 libras.

Salvador Merino, 17/2/1993.

UNA DE VENTA POR CORREO

Hacia finales de Enero del glorioso 1992 pedí a TK Computerware (en adelante TK) los dos clásicos "Assembly language programming" y "QDOS compannion" y el logicial "QPAC 2". Para el pago de 9 libras por cada libro y 35.20 + 9 libras por los programas di mi número de tarjeta VISA.

Sorprendentemente, antes de cumplirse un plazo de quince días recibí vía un servicio courier un paquete que contenía el primero de los libros, su correspondiente albarán y una nota en la que decían que "QDOS compannion" estaba agotado y próximamente me enviarían QPAC 2.

Yo en ese momento tenía preparada una carta-pedido a Digital Precision por "LIGHTNING Special Edition" (vengo utilizando la V1.22 desde 1989) y, deslumbrado por la actuación de TK y puesto que el pedido no estaba completamente suministrado, pensé que sería preferible solicitarles a ellos una ampliación de pedido y de esta forma la relación comercial sería con una única casa. Así lo hice. Y así me fue:

A finales de Abril Visa me notifica un cargo de TK por 53,20 libras (el libro más QPAC 2 con sus gastos de envío) fechado 10-2.

El 11-3 envío a TK una carta explicando que no he recibido los logiciales y dándole la alternativa de saldar el primer pedido con el envío de QPAC 2 o enviar los dos logiciales y efectuar el cargo suplementario por LIGHTNING S.E.

A últimos de Marzo Visa me notifica que TK me ha cargado 88,20 libras el 25-2 (los dos logiciales más gastos de envío). El 3-4 telefoneo a TK y me dicen que me contestarán una vez consultados sus registros.

El 13-4 recibo carta de TK fechada el 2-4: Ambos logiciales han sido enviados por carta en correo aéreo el pasado 2 de Marzo.

El 19-4 recibo contestación a mi llamada: mi carta del 11-3 fue interpretada como un segundo pedido, cargándome 88,20 libras. Me devolverán 35,20 libras (con lo que me están cobrando 44 + 9 libras por LIGHTNING S.E.)

El 30-4 Visa notifica la devolución de las 35,20 libras.

No habiendo recibido nada, el 29-5 escribo nuevamente a TK comunicándoles este hecho y quejándome de que por un envío están cobrándome dobles gastos.

El 30-6 Visa notifica un nuevo cargo de 88,20 libras con fecha 12-6. Nueva carta a TK.

El 21-9, no habiendo tenido noticia alguna, escribo nuevamente a TK: Protesto por los cargos indiscriminados y adjunto una relación cronológica de los hechos hasta el momento. Por otra parte reclamo a Visa la retrocesión del cargo de 88,20 libras de fecha 12-6 y envío a QL World, The troubleshooter, una carta exponiendo la relación con TK.

El 22-10 recibo QPAC 2 sin albarán ni comentario alguno.

El 2-11 acuso recibo a TK de QPAC 2 y reclamo el envío de L.S.E., así como justificación de las cantidades cobradas hasta el momento. También comunico a QL World la recepción de QPAC 2.

Unos días más tarde Visa me notifica la retrocesión del último cargo de 88,20 libras de fecha 12-6.

El 31-12 Visa me notifica el pago de un nuevo cargo de 96,24 libras (¿?) con fecha 12-6 y referencia 7-12. En el mismo día reclamo a Visa la retrocesión del mismo.

A mediados de Octubre había empezado a escribir para QLíper un articulito sobre esta saga, pero el desencadenamiento de los acontecimientos me obligó a rehacer sucesivos borradores y finalmente, lo que iba a ser un breve resumen, para mantener la hilación de fechas y circunstancias, se ha convertido en este peñazo. Y, lo que es más, queda sin acabar y a falta de dar un paso decisivo respec-

to de Visa y, por supuesto, TK. Es mi bolsillo, qué demonio. En cualquier caso creo que queda suficientemente clara la moraleja acerca de los peligros que entraña la utilización de tarjeta de crédito en transacciones de esta naturaleza.

25 de Enero de 1993 Miguel Estarellas (91) 5 73 07 01

C68: UNA PREGUNTA

En mis primeros pinitos en C he chocado de frente con la función ATOF, para convertir una cadena en un número racional en doble precisión. Mi problema concreto es que al utilizarla como argumento en un procedimiento que pone valores en una pila, éste no funciona, la pila no recibe los valores. C68 ha realizado la compilación sin errores ni llamadas de atención. Es más, todo funciona correctamente si hago las correspondientes declaraciones de variables y procedimientos como enteros y utilizo la función ATOI. ¿Tiene alguien alguna experiencia en esta función? ¿Es un "bug" de C68? ¿Merece la pena hablar con D. Walker?

26 Enero 1993 Miguel Estarellas (91) 5 73 07 01

MIRACLE GRAPHICS CARD: Algo más que una simple tarjeta gráfica

En enero de 1992, Miracle Systems hizo una demostración de su tarjeta gráfica en la reunión internacional de usuarios de QL que se celebró en Italia. Ha pasado un año desde aquel día y la GRAPHICS CARD ha sufrido muchas modificaciones. Los rumores que circulan hacen pensar que la GRAPHICS CARD final tiene poco que ver con su nombre. Sus características actuales son:

- Viene en una caja con fuente de alimentación y espacio para nuevas tarjetas y un par de disquetteras de 3.5".
- Interface para teclado PC.
- Interface para monitor VGA.
- Interface serie.
- Interface paralelo.
- Interface SCSI.
- Sistema operativo SMS-2 de Tony Tebby.
- Selector modo VGA o RGB.
- Imprescindible la GOLD CARD.

Dadas las características del nuevo producto. Aunque no está confirmado, me hace pensar que GRAPHICS CARD + GOLD CARD podría funcionar sin necesidad de la tarjeta madre QL. Creo que ya entiendo por qué Miracle Systems dice que la GRAPHICS CARD es la base de la nueva generación de QLs.

Salvador Merino, 19/1/1993.

IMPRESORA DE CHORRO DE TINTA NETSET JS_160

En una visita a Sinclair Store he tenido la ocasión de ver esta impresora que está en oferta y me ha parecido interesante. Resumo: Es Olivetti bajo otra marca, 160/80 cps, 300x300 ppp, emulación HP Deskjet +, interfaz paralelo. Es de pequeño tamaño. La oferta incluye alimentador automático de hojas: 49900+IVA.

27 Enero 1993 Miguel Estarellas (91) 5 73 07 01

EL RINCON DEL PINCHA-DISCOS

¡ Hola a todos !

A aquellos que seáis nuevos os advierto que QLíper posee una colección o compilación de software PD o ShareWare QL muy voluminosa (alrededor de 1.500 a 2.000 programas, y abundante material diverso en disco). La lista de discos disponibles ha sido editada en discos QLíper atrasados (QLíper 36-41, o CUQ

1-35). De todas maneras, puedo facilitar una lista en disco si me enviáis un disco formateado y un sobre con vuestra dirección y debidamente franqueado para su retorno.

Las condiciones de acceso de la librería PD QLíper siguen siendo las mismas. Hay dos opciones:

a) Hay que enviar lo siguiente:

- Discos 3.5" formateados a 1440 sectores sin errores.
- Un sobre adecuado para discos con vuestra dirección y debidamente franqueado para su retorno. (Pesad los discos y el sobre, y consultad la tarifa de correos).
- 50 pesetas por cada disco a copiar para cubrir gastos de grabación y adquisición de nuevo material.

b) Solamente hay que enviar 225 ptas por disco a copiar. En el precio está incluido el disco, la grabación, el sobre y los sellos.

Para cualquier consulta, podéis llamar a los teléfonos (95) 2475043 (trabajo) y (95) 2474887 (casa).

Salvador Merino, 2/2/1993.

NOVEDADES EN LA LIBRERIA QLIPER

Las nuevas adquisiciones ya disponibles son:

- Librería QPTR v2.0.- Sustituye a la anterior v0.5. Posee una utilidad para trabajar con el EasyPointer.

- Librería CURSES.- Es una librería para escribir programas basados en pantallas de caracteres. La implementación y facilidades son las definidas en el UNIX SVR4.

- Librería CPORT.- Esta librería ha sido diseñada para usar con el programa CPORT (D.P.) y el compilador C68. Sin embargo, pueden usarse directamente todas sus funciones en nuestros programas. Esta librería contiene la mayoria de los comandos SuperBASIC, TK2 y TURBO (y un par del QLiberator). Si alguien sabe programar en SuperBASIC con TK2 Toolkit y TURBO Tookit, y desea programar en C sin aprender nuevos comandos, esta es su librería de funciones.

- ELVIS Editor.- Es un clono de vi/ex, el editor estándar UNIX. Aún no he podido detenerme en leer la documentación (85 páginas). Solamente puedo decir que no estoy muy seguro de si esto es un editor o un sistema operativo de los años 70. Es muy probable que posea más comandos y opciones que The Editor Special Edition (D.P.).

- GNU Text Utilities (2 discos).- Son una colección de utilidades relacionadas con el tratamiento de ficheros de textos. He estado mirándolas por encima, y creo que hay de todo lo imaginable. Eso sí, son utilidades tipo sistemas operativos en la línea del CP/M, MS-DOS, y UNIX, lo que significa comandos tipo ficheros en disco que hay que pasarles una cantidad de opciones o parámetros impresionantes.

- QPACER.- Es una utilidad para aquellos que tengan dificultades para crear el programa BOOT para QPAC2. Para mayor información, ver revista Sinclair QL World Noviembre 1992 páginas 12 y 13.

- ZM1+ Spectrum Emulator.- Se trata de la versión ShareWare del emulador Spectrum realizados por los italianos. Se adjunta juegos como JETPAC, ATIC ATAC y PSST. Todos muy jugables en un QL a 24 MHz.

- Text 'N' Graphics (Demo).- Demostración de un programa comercial editado por DJC. Permite introducir pantallas dentro de ficheros de texto.

- C programming Tools 1.- Contiene las herramientas de programación Cproto, Indent y Unproto.

- C Debugging Tools.- Contiene las librerías para depuración de errores: Lib Debug y Lib Malloc.

- Compilador C68 v3.05.- Ultima versión. Se necesita la v3.04 o superior para poder utilizar cualquier librería C68 nombrada anteriormente.

- Emulador QL en el AMIGA v3.10.- Solamente puedo copiar el segundo disco en formato QDOS. Esta versión es la penúltima del emulador. La mayor diferencia frente a la v3.03 es que los discos formateados o escritos en el AMIGA pueden ser leídos en cualquier QL. Este programa es el único que no se puede copiar, pues no tengo un Commodore AMIGA a mano, pero digamos que es el único programa que puedo prestar con la condición de que me lo devuelvan.

Salvador Merino, 2/2/1993.

EL NUEVO EMULADOR DE QL PARA EL COMMODORE AMIGA
===

SEGúN ME COMUNICA SALVADOR MERINO, HA APARECIDO EN ESCENA UNA NUEVA VERSIóN DEL EMULADOR DE QL PARA EL COMMODORE AMIGA. Esta parece que soluciona los problemas de manejo de disquete que tenía la antigua versión. Lo cual me hace estar ansioso por conseguirla.

Respecto a la versión antigua, que es la que sigo utilizando, decir que funciona perfectamente con el nuevo sistema operativo el AMIGADOS versión 2, salvo el pequeño problema con definiciones de pantalla mayor a la estándar de 640x400, posibilidad que soporta bien AMIGADOS pero no el emulador. Este hecho me permite presuponer que lo hará correctamente con los nuevos AMIGA 1200 y 4000 recién aparecidos en el mercado y ya disponibles en España. Dado que creo que os será de interés os comentaré algunas de las características de estos nuevos ordenadores.

Dos son las más fundamentales: arquitectura de 32 bits y nuevo juego de coprocesadores chipsets tambien de 32 bits con gráficos de 24 bits. El Amiga 1200 viene con un 68020 a 14 MHz y 2 MB de memoria chip y el Amiga 4000 con un 68040 a 28 MHz. Aparte de la CPU destaca su nuevo juego de chips con los que es capaz de trabajar con pantallas de resolución variable de 200x200 a 1480x1480 pixels con 256 colores reales simultáneos en pantalla en cualquier resolución o hasta 16 millones de colores en un nuevo modo HAM ideal para vídeo. En modo HAM no se permiten cambios bruscos de color de un pixel a otro, han de ser variando la gama de colores con pixeles intermedios de colores mixtos lo cual suaviza los contornos de un modo natural. Dado que la nueva arquitectura de los chips es de 32 bits en su conjunto, aparte de más capaces son 4 veces más veloces en sus operaciones, puesto que estos chips antes eran de un bus de direcciones de 9 bits reales que llegaba a los 18-20 bits por multiplexación Los precios dependiendo del modelo son más caros o más baratos en España que en Inglaterra o en USA. Así, el Amiga 1200 vale 89000 pts o 119000 con disco de 70 megas y el Amiga 4000 vale 310000 con disco de 52 Mb o 370000 con disco de 120 Mb (399 y 2000 libras en Inglaterra), y se espera que en los próximos meses aparezcan versiones más baratas del 4000 con CPU 68E030 o 68030 y un nuevo super-AMIGA con aparición simultánea a la del anunciado Motorola 68060 (¿Qué fue del MC68050?). Los antiguos modelos 1000 500 500+ 2000 y 3000 tenderán a desaparecer, aunque su arquitectura permanecerá viva con el Amiga 600, lo cual presupone que el software habrá de ser sobre todo compatible, lo cual está garantizado si se siguen las directrices del AMIGADOS, que proporciona ya desde la versión 2 rutinas capaces de adaptarse óptimamente al tipo de máquina, su CPU y sus coprocesadores aritméticos o de DMA.

En el Amiga 4000 siguen manteniéndose las ranuras del bus ISA de AT, permitiendo albergar las tarjetas BridgeBoard, las cuales han bajado de precio de forma espectacular. Así se puede disponer de un AT en multitarea con AMIGAdos por 18000 pesetas, de un 386sx 25 MHz con 2 Mb por 60000 y de un 486SLC con 8 Mb por unas 100000. con sus slot de coprocesador, controladores de floppy de 2.8 Mb y de disco duro IDE. También existen placas para ejecutar software de Macintosh que ofrecen ranura de expansión de Mac por 55000 pts, permitiendo ejecutar sin problemas el system 7, y emuladores software para Atari ST, BBC, Commodore 64, Spectrum, CP/M y QDOS. Asimismo se puede conseguir el Unix System V versión 4 como sistema operativo El sonido de estos nuevos Amiga sigue siendo el mismo: 4

canales de sámpler estéreo de 8 bits, aunque el nuevo super-Amiga contará de serie con lo que hasta ahora se ofrece sólo como placa de expansión: 8 pistas estéreo de sonido digital de 16 bits, (1 compact disc consta de 2 canales de 12 bits). Las placas disponibles junto con sincronizador SMPTE y software de grabación directa a disco duro y de efectos de sonido valen en España 55000 pesetas, la de 4 pistas de 12 bits, y algo más del doble la de 8 pistas de 16 bits, aunque si se comprara en USA los precios serían casi la mitad incluyendo impuestos y gastos de envío.

Nacho Enrique (19/1/1993)
C/ Miguel de la Iglesia 3, 1-C
28027 Madrid
Teléfono (91) 3773915

LA QXL

Hasta hoy sabía que Miracle Systems estaba trabajando en una tarjeta QL para PC, pero cuando he visto hoy la primera foto del QXL a escala (más pequeño que la GOLD CARD) en la revista Sinclair QL WORLD de febrero del 93, mi mente ha salido de dudas: el mundo QL tiene su futuro garantizado.

¿Qué es la QXL? Es una pequeña tarjeta (media tarjeta, dicen los anglosajones) que convierte al PC en un QL compatible.

¿Qué contiene la QXL?

- 68EC40 20 MHz.
- 8 megabytes de RAM.
- Puerto Network compatible con el QL.
- Sistema operativo compatible con QDOS, de Tony Tebby.
- Usa el PC como un sistema de entrada y salida.

¿Qué configuración mínima de PC necesita?

Un XT o compatible, 512K, EGA y un slot libre.

Según mi opinión, la tarjeta es tan pequeña que es muy probable que algunos PC portátiles con slot de expansión posean suficiente espacio en su interior para alojar la QXL. Si fuese cierto, habría nacido el primer compatible QL portátil.

¡Larga vida al QDOS!

Salvador Merino, 22/2/1993

PROGRAMAS C68 CONVERTIDOS A TURBO C MS-DOS

Doy autorización de uso de todos los programas y rutinas 'C' escritas en un primer momento en C68 para QDOS en PD, y luego convertidas a TURBO C con muy pocas modificaciones.

Advertencia: no recuerdo si son las versiones TURBO C v1.00 o TURBO C++. Si son las primeras, es necesario tener cargado el ANSI.SYS (el driver de pantalla de ANSI). Todos estos programas fueron escritos y compilados usando PC CONQUEROR normal y gold.

Mi interés por escribir programas comerciales se ha esfumado. El listón está hoy en día demasiado alto. Prefiero escribir software PD o ShareWare.

Salvador Merino, 22/2/1993.

ACTIVIDADES EMPRESARIALES. REGISTRO DE INGRESOS O VENTAS

Para no perder la práctica uno tiene que escribir algo de vez en cuando. Este programa en particular es del tipo de aquellos que prácticamente no sirven casi

para nada, pero por lo menos ordena por fechas y no suele equivocarse en las sumas. Emula un libro de ACTIVIDADES EMPRESARIALES de Registro de Ingresos o Ventas a la perfección. La salida por impresora es en el mismo formato página del libro emulado (posee las mismas líneas por página).

El programa está basado en las mismas rutinas que he usado en programas anteriores, pero ahora están más depuradas y probadas que antes, por lo cual tengo que reconocer que el esfuerzo realizado en el desarrollo del programa ha sido mínimo (probablemente menos de 8 horas), ya que la mayor parte del programa estaba escrito y con garantías absolutas de correcto funcionamiento.

El programa se ejecuta con:

```
exec flp1_actividades_ventas_exe
```

Lo primero que pregunta es por el nombre del fichero. Podría ser:

```
flp1_ventas_1_93
```

Esto intentaría abrir dos ficheros:

1) flp1_ventas_1_93_fxi
2) flp1_ventas_1_93_fxd

Y si por una casualidad no existen, crearía esos ficheros como nuevos.

La pantalla de presentación de datos se ha cuidado bastante añadiendo gráficos (líneas) y optando por un menú de cursores. Aunque creo que se podría refinar un poquito más.

Una vez dentro del Editor de Formularios (opciones Añadir o Modificar) las teclas de control (ir a registro anterior, terminar, etc...) son las mismas que habían sido definidas durante el cursillo "Practicando con el C68". Por lo tanto, volverlas a repetir lo veo innecesario.

Con este programa se adjunta otro que es un editor de lista, el cual aún no tiene utilidad, pues la v1.00 de Registro de Ingresos o Ventas no posee aún la opción que debería procesar la lista de datos. Añadirla es sencillo, hay dos maneras posibles:

1) Ponerla como pregunta después de pedir el fichero Ventas.
2) Ponerla como opción en el menu de cursores.

La única pega que tiene la opción (1) frente a la (2) es que solamente procesa un ficheros mientras no entremos en el menú de cursores. En ambas opciones la rutina sería la misma. Se utilizaría las rutina AÑADIR ligeramente modificada. Por ejemplo, en vez de usar la rutina EDITOR DE CAMPOS utilizaríamos una nueva que leería la lista de datos (esta rutina es la misma que utiliza el programa Editor de Lista para leer datos de disco a memoria).

El próximo programa que debería escribir sería REGISTRO DE GASTOS Y COMPRAS. Ese programa sería casi idéntico que este, pero con la novedad de un nuevo campo que debería indicar a qué tipo de gasto pertenece.

Salvador Merino, 20/1/93.

WHISHES / DESEOS

Now that a new year begins I think that it is a good time to make a wish list of all the things I would like to have for the QL.

SOFTWARE.

First of all, I must say that I am not a good programmer, but I would like to encourage all the good programmers that could read this to put their efforts in order to make something similar to the things I am proposing, as I think that could be interesting to a large number of QL users.

1) A new spreadsheet. Spreadsheets are one of the most popular pieces of software, no matter which computers you refer to. However, there is a lack of decent spreadsheets for the QL. ABACUS is easy to work with, easy to learn, but it is very slow and inadequate for today's standards.. I would like to see in the public domain a spreadsheet with the following specs.:

a) It must be as fast as possible, so it should be written in C or in assembly.

b) It should be user-friendly, preferably with the same structure of ABACUS or something like MULTIPLAN (MS-DOS systems).

c) It should be able to export and import from/to ABACUS, and from the most popular format in the PC world, the LOTUS 1-2-3 one (.WKS).

d) It is not necessary to be a sophisticated program, I mean, no need for complicated graphics (after all, we have EASEL).

2) What about a ATARI GEM emulator?

It could sound funny, but it is not as difficult as it seems. The GEM environment of the ATARI ST, which takes care of the Windows, Icons and Pull-down menus, is not an Operating System developed specifically for the ATARI ST computer (in fact it is only a part of the complete O.S. of the ST).

There is a version of GEM for the PC compatible computers, but it did not really catch on, due to the success of the similar front-end WINDOWS.

O.K. the GEM O.S. does not have the nicest icons or the nicest colours on screen, but it is simple and very ease to work with, much more than the WINDOWS environment. It does not need of a large hard disk to work, as WINDOWS does, and works comfortably on a 520 kbytes machine.

On a ST computer it is included in the ROM, but it could be easily loaded from a floppy disk (the first ATARI computers did not have the GEM built-in ROM but you have to load it from floppy, as the MS-DOS).

Most of GEM on the ST is written in C, so it is not very fast. Making an GEM emulator written in assembly, you could achieve a faster implementation on the QL, in spite of its 8 bit data bus and its slower clock frecuency (7.5 MHz instead of 8 MHz), although, of course, the programs will run slower than on an ST (on their normal operation). With a GOLD CARD the speed would be very good, twice of a normal ST.

If DIGITAL PRECISION managed to make a very good MS-DOS emulator, I think that a GEM emulator should be easier to make. That way, all the well-behaved GEM programs available for the ATARI ST could be used for in the QL.

Bear in mind that the GEM programs (the well written ones) could work easily in a lot of configurations, from 320x200 pixels to 1024x760 pixels, or more, screens, with 2 from 16 million colours. All the programs that do not have to relay in some hardware peculiarity (for instance, the MIDI ports) could run on a QL without problems (at least in theory, as far as I know). I do not know if there will be problems with the copyright laws, but I would like that someone will write this emulator.

HARDWARE.

1) A kind of GOLD CARD but with a 68030 processor, running at 20 MHz, at least, and 4 Mbytes of RAM, with a price below 500.

2) A cheap graphic card, preferably connected internally on the QL (is there room for doing that on a QL?), or connected to the ROM port. A video RAM of 64 Kbytes and a palette of 4096 colours could be enough. This way, a 320x200 pixels with 256 colours, and a 640x480 with 2 colours, screens will be available, that could allow to implement new MS-DOS emulations (VGA low and monochrome resolutions) for the CONQUEROR emulator from DIGITAL PRECISION, opening the use of a vast number of MS-DOS applications. The combination of this card and the new GOLD CARD 68030 based would be superb!

3) And SCSI interface. With this I/F and the suitable software you could access

to a vast number of new and wonderful devices, as hard drives, CD ROM drives, flopptical drives, and so on.

4) An IDE I/F, to connect the newest and less expensive hard drives for the PCs.

5) And finally not a wish, but a dream: a new QL clone, built around a 68040 processor, with lots of free slots and with and affordable price. Will we see that someday?

Ahora que un nuevo año comienza creo que es un buen momento para escribir una especie de carta a Los Reyes Magos con todo aquello me me gustaría ver para el QL.

PROGRAMAS.

Antes de nada,debo decir que no soy un buen programador, pero me gustaría animar a todos los buenos programadores que puedan estar leyendo esto para que dirigan sus esfuerzos para hacer algo similar a las cosas que propongo, ya que creo que pueden ser interesantes para una gran cantidad de ususarios del QL.

1) Una nueva hoja de cálculo. Las hojas de cálculo son uno de los programas más populares, para cualquier ordenador de que se trate. Sin embargo, hay se hecha en falta una hoja de cálculo decente para el QL. ABACUS es sencilla de aprender y de utilizar, pero es muy lenta e inadecuada para los estándares de hoy en día. Me gustaría ver en el dominio público una hoja de cálculo con las siguientes características:

a) Debe ser lo más rápida posible, luego debería estar escrita en C o en ensamblador.

b) Debería ser "amistosa", preferiblemente con la misma estructura del ABACUS o algo como MULTIPLAN (para el MS-DOS).

c) Debería ser capaz de exportar e importar a/desde ABACUS, y a/desde el formato más popular en el mundo del PC, el formato LOTUS 1-2-3 (.WKS).

d) No es necesario que sea un programa sofisticado, no hace falta que sea capaz de presentar complicados gráficos (para eso tenenemos el EASEL).

2) ¿Que tal un emulador del GEM del ATARI?

Puede sonar extraño, pero no es tan difícil como puede parecer. El entorno GEM del ATARI ST, que se encarga de gestionar las Ventanas, Iconos y Menús desplegables, no es un sistema operativo desarrollado específicamente para el ATARI ST (de hecho, es sólo una parte de todo el S.O. del ATARI ST).

Existe una versión del GEM para el PC compatible, pero que no tuvo gran éxito, debido al éxito del entorno similar WINDOWS. De acuerdo, el GEM no presenta en pantalla los iconos o los colorines más bonitos, pero es sencillo y muy facil de utilizar, mucho más que el entorno WINDOWS. No precisa para trabajar de un disco duro de gran capacidad, al contrario de WINDOWS, y trabaja comodamente en una maquina con 512 Kb de memoria RAM.

En un ordenador ATARI ST está incluido en la memoria ROM, pero se puede cargar facilmente desde disco (los primeros ordenadores ATARI no incorporaban en ROM el GEM, sino que tenían que cargarlo desde disco, como el MS-DOS).

La mayor parte del GEM en el ST está escrito en C, por lo que no es muy rápido. Implementando un emulador del GEM escrito en ensamblador, se podría disponer de una versión más rápida en el QL, a pesar de su bus de 8 bits y su velocidad de reloj más lenta (7.5 MHz en lugar de 8 MHz), aunque, por supuesto, los programas correrían más lentos que en un ST (en su funcionamiento normal). Con una GOLD CARD la velocidad sería muy buena, el doble de la de un ST estándar.

Si DIGITAL PRECISION se las arreglaron para hacer un buen emulador del MS-DOS, creo que un emulador del GEM debería ser más sencillo de implementar.

De este modo, todos los programas bien escritos bajo el entorno GEM disponibles para el ST podrían ser usados en el QL. Hay que tener en cuenta que los programas GEM (los escritos siguiendo las reglas) pueden trabajar cómodamente en

una gran cantidad de configuraciones, desde pantallas de 320x200 a 1024x760 o más pixeles, con de 2 hasta 16 millones de colores. Todos los programas que no tuvieran que basarse en una particularidad del "hardware" (por ejemplo, los puertos MIDI) podrían correr en el QL sin problemas (al menos en teoría, hasta donde yo sé). No sé si existirían problemas con las leyes del "copyright", pero me gustaría que alguien escribiera este emulador.

HARDWARE.

1) Una especie de GOLD CARD pero con un procesador 68030, funcionando a 20 MHz por lo menos menos, con 4 Mbytes de RAM y un precio inferior a 100.000 pts.

2) Una tarjeta gráfica económica, preferiblemente conectada en el interior del QL, o al puerto ROM. Sería suficiente una menoria de vídeo de 64 Kbytes y una paleta de 4096 colores. De este modo se dispondría de pantallas de 320x200 pixeles con 256 colores, y 640x480 con 2 colores, las cuales permitirían implementar nuevas emulaciones MS-DOS (baja resolución VGA y resolución monocroma VGA) para el emulador de CONQUEROR de DIGITAL PRECISION, abriendo el uso a un amplio número de aplicaciones MS-DOS. La combinación de esta tarjeta gráfica y de la nueva GOLD CARD sería explosiva.

3) Una interfase SCSI. Con esta I/F y el "software" adecuado se podría acceder a un vasto número de nuevos y maravillosos dispositivos, como discos duros, discos CD ROM, discos ópticos (grabables y borrables, como si se trataran de discos magnéticos convencionales), etc.

4) Una I/F IDE, para conectar los más recientes y económicos discos duros para los PCs.

5) Y finalmente no un deseo, sino un sueño: un nuevo compatible QL, constituido alrededor de un procesador 68040, con montones de ranuras libres y con un precio asequible. ¿Lo veremos algún día?

Javier Zubieta Aguirre
6/1/93

CIFRAS
======

Presentación

Bueno, por fin os puedo enseñar este programa. Tenía muchas ganas de hacerlo, pero no había podido compilar el programa con C68. Ahora, gracias a Dave Walker, he podido hacerlo. Una confesión: llevo 17 años programando y este es el programa que más me gusta de todos los que he hecho. Espero que también os guste a vosotros.

Este programa se presenta como resultado del estudio de la asignatura "CONCEPTOS FUNDAMENTALES II", impartida por Martín Molina (de "EQ sistemas inteligentes") en el curso "Sistemas expertos en la empresa" impartido en el CETICAM durante el año académico 1991-92.

El programa ha sido implementado utilizando la idea directora ofrecida por el profesor Martín Molina.

El programa resuelve un problema real que se presenta en el concurso televisivo "Cifras y letras" utilizando técnicas de inteligencia artificial de búsquedas en espacios de estados. El enunciado del problema es:

> A partir de 6 números obtenidos del conjunto {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 25, 50, 75, 100} (se pueden repetir) hay que obtener otro número natural perteneciente al intervalo [100,999]. Se pueden utilizar las cuatro operaciones elementales entre los números y cada número sólo puede ser utilizado como máximo una vez. El ganador es el concursante que más se aproxima al número (por defecto o por exceso).

El programa implementa una búsqueda por un árbol de estados que tiene un número máximo de aproximadamente 3 millones y medio de nodos.

Derechos

Versiones realizadas

Hasta llegar a la versión que os presento, tuve que hacer muchas pruebas. Esta es la historia de esas pruebas.

CIFRASP1_C: Es el primer propotipo. Realiza una búsqueda exhaustiva, por lo que el tiempo para el caso real es muy elevado.

CIFRASP2_C: Segundo prototipo. No muestra en pantalla los nodos que se van explorando. Se obtiene un aumento del rendimiento de un 50%. La solución final se presenta más limpia, quitando los paréntesis alrededor de los números.

CIFRASP3_C: Tercer prototipo. Las operaciones se almacenan en notación polaca inversa. Se utilizan separadores entre números y operaciones. Se obtiene un aumento del rendimiento de un 35%.

CIFRASE1_C: Primer experimento. Se intenta averiguar cuántos nodos con 4 elementos hay en el árbol de búsqueda y cuántos son distintos.

CIFRASE2_C: Segundo experimento. Igual que el primero, pero no se muestra en pantalla los nodos que se van explorando. Sorprendentemente, el programa tarda el triple, por lo que se desecha la idea.

CIFRASE3_C: Tercer experimento. Igual que el primero, pero los nodos almacenados están en orden decreciente. El tiempo empleado es un 30% que el experimento 1.

CIFRAS1_C: Primera versión. Utiliza el prototipo 3 junto con la idea del experimento 3. Es decir, los nodos con cuatro elementos sólo se expanden cuando aparecen por primera vez; luego se podan. En pantalla va apareciendo el número de estos nodos que se han generado y al final del programa, cuántos son distintos.

CIFRAS2_C: Segunda versión. Mejora general de detalles de implementación. El nodo de partida se coloca en orden decreciente.

CIFRAS3_C: Tercera versión. Los nodos con 2 elementos reciben un tratamiento especial: no se generan los caminos de las operaciones con ellos a no ser que mejoren la aproximación. Se mejora el rendimiento de forma que tarda en el caso peor un tercio que la versión anterior.

CIFRAS4_C: Cuarta versión. La función que examina los nodos con 2 elementos ha sido mejorada. La mejora es apreciable, pero no espectacular.

CIFRAS5_C: Quinta versión. Se pulen pequeños detalles.

CALC1_C: Permite convertir una expresión RPN en algebraica.

CALC2_C: Permite convertir una expresión RPN en algebraica explicando cada operación.

CIFRAS6_C: Sexta versión. La solución se da explicando cada operación.

Adaptación al QL

Este programa se empezó a escribir en un PC portátil con la intención de que en esa máquina llegara a resolver el problema en tiempo real, es decir: en los 45 segundos que se ofrecen en TV. Una vez conseguido eso, dejé de preocuparme por la velocidad de ejecución. Al transportar el programa al QL me he encontrado con una gran diferencia de rendimiento, lo que simplemente demuestra la necesidad que tiene el mundo QL de hardware nuevo (afortunadamente, la tarjeta QXL es ya una realidad). Por tanto, veréis que en vuestros ordenadores algunos problemas

tardan más de esos 45".

Llevar el programa al QL ha sido muy sencillo, ya que está escrito en ANSI C, con 2 pequeñas excepciones: dos funciones específicas de Turbo C que tienen un equivalente en C68. Compiló fácilmente (hace mucho tiempo), pero ¡ay! falló al montar. El programa decía que no tenía memoria. Bueno, tengo una Gold Card, así que le di 1 mega: mismo error, falta de memoria. ¿Entonces qué pasa? Al fin pude escribir a Dave Walker (el coordinador del equipo que desarrolla C68) quien muy amablemente me explicó que debía usar las opciones -bufp o -bufl, que asignan memoria a dos zonas distintas del montador. ¡A la primera! Si queréis recompilar el programa debéis usar la opción -bufp.

Para ver el programa en acción basta teclear LRUN FLP1_Cifras_bas

Pedro Reina, X.3.3.1993

OLIMPO

Según os contaba en otro número de QLíper, estoy escribiendo un método de programación orientado al objeto con la intención de facilitar la vida del sufrido programador.

Tengo una versión en SuperBASIC que ha sido absolutamente arrollada por la que he escrito en C. Voy a concentrar mis esfuerzos en ella, ya que está más adelantada.

Podéis ver en el disquete un programa de demostración de las posibilidades de Olimpo, junto con el código del programa. Quiero resaltar que exactamente el mismo código ha sido compilado en un PC, lo que abre muchas posibilidades.

Para ver la demostración basta teclear

LRUN FLP1_Olimpo_Demo_bas

Es conveniente tener Toolkit II instalado, ya que si no, una de las posibilidades no funcionará completamente.

Si alguno quiere tener el sistema, me lo puede pedir a mí o a Salvador, ya que está en la biblioteca de QLíper. Especificad si queréis sólo la versión QL o también la PC.

Para que os hagáis mejor una idea del sitema, incluyo a continuación parte de la documentación:

Presentación

El sistema Olimpo pretende facilitar la tarea del programador en C en varios aspectos:

Aumentar la legibilidad del código.
Acortar el tiempo de desarrollo de aplicaciones.
Reutilizar código.
Facilitar el transporte entre plataformas.

Por todo ello, he decidido implementar el sistema orientado a objetos. Comienzo con objetos de bajo nivel, que se comunican directamente con el hardware y hay que reescribir cuando se intenta llevar el sistema a una plataforma distinta. Estos objetos permiten uniformizar el tratamiento de las funciones de bajo nivel y sobre ellos se puede construir objetos más abstractos que dan servicios más complejos. En un futuro, pretendo crear objetos de más alto nivel, que se encarguen ellos mismos de parte de la gestión del programa.

De momento Olimpo está disponible para el QL usando el compilador C68 (v3.01) y para PC usando Turbo C. Pretendo escribir una versión sobre UNIX, usando COHERENT.

El sistema es de dominio público. Se puede distribuir libremente siempre y

cuando se copie completo. Cualquier sugerencia, comentario o pregunta será bien recibido. Estos son mis datos:

Pedro Reina
c/ Sandoval 6
28010 Madrid
España

Teléfono: 91 - 593 81 59

Observación:
A lo largo de la documentación se utiliza como separador entre el nombre de un fichero y su extensión el carácter ".", el habitual en UNIX y MS-DOS. En el caso del QDOS, el separador que realmente se utiliza es "_".

Descripción del sistema

El sistema está distribuido por objetos. Cada uno se ocupa de una parte de la programación, y he procurado hacerlos lo más independientes entre sí que podido.

Hay un fichero de definiciones generales, en la que se tiene:

El tipo de datos "lógico", con sus valores SI y NO.
El tipo de datos "contador", con un rango de valores de -32768 a 32767.
El tipo de datos "entero", que es un alias de "int".

La constante NULO, que es un alias de NULL (carácter nulo).
La constante NIL, que es el 0 (se utiliza en LISP como fin de lista, y en ese sentido lo voy a usar en Olimpo).

Las funciones Abs(x), Max(x,y) y Min(x,y).

Cada objeto tiene un identificador de tres letras y todas las funciones y macros de ese objeto comienzan con ese identificador.

A continuación describo cada objeto:

Pantalla

Este objeto permite controlar la apariencia completa de la pantalla. La función Pan_Define() permite iniciar los valores por defecto y la función Pan_Cierra() restaurar todo para volver al sistema operativo.

La pantalla es de 24 filas numeradas de 0 a 23 comenzando por arriba y 80 columnas numeradas de 0 a 79 empezando por la izquierda.

Se dispone de 4 colores, definidos con los macros BLANCO, NEGRO, ROJO y VERDE. Se manejan con las funciones Pan_Papel(), Pan_Tinta() y Pan_Color().

Es posible controlar la posición del cursor y si se debe ver o no.

Se puede escribir en la pantalla texto (cadenas), enteros de cualquier rango y caracteres. Hay funciones que colocan el cursor y luego escriben el texto, entero o carácter. Se puede activar o desactivar el modo de escritura resaltada.

Se puede borrar la pantalla completa, una línea o una zona rectangular.

Azar

Se pueden obtener números enteros aleatorios en el margen que se desee. Basta iniciar el generador al principio con Azr_Inicia() y luego ir pidiendo enteros con Azr_Entero().

Tiempo

Para medir intervalos de tiempo se puede usar Tim_Crono() y para establecer pausas de una duración determinada, Tim_Espera().

He definido el tipo de datos "tiempo", que es el que devuelve Tim_Crono().

Cadena

En ANSI C se definen gran variedad de funciones para manejar cadenas. Básicamente este objeto utiliza esas funciones con un nombre más claro. También hay otras pequeñas funciones.

He definido el tipo "cadena", que es simplemente un vector ("array") de 80 caracteres (más uno para el carácter NULO). Es muy útil para nombres de ficheros y para cadenas que van a aparecer en la pantalla.

Sonido

Este objeto se encarga de las señales acústicas. Se pueden conectar con Son_Enciende(), desconectar con Son_Apaga() o cambiar de estado con Son_Cambia().

Las llamadas a las funciones de este objeto no provocan inmediatamente la generación del sonido, eso dependerá de si en ese momento está conectado o no.

Tecla

He definido el tipo "tecla" y la función Tec_Pulsada() como básicos para poder olvidar las grandes diferencias en el tratamiento de las teclas en los diversos sistemas operativos. En Olimpo sólo hay que saber que Tec_Pulsada() devuelve la tecla que haya pulsado el usuario, que pertenece al tipo "tecla". Junto con eso, hay toda una colección de macros que definen cada tecla; por ejemplo, la tecla F1 está definida con el macro TEC_F1, y el programador no necesita conocer qué codigo (o códigos) asigna el sistema operativo a F1 (cada uno lo hace de una forma distinta). En esta versión no dispongo de macros para la ñ ni las vocales acentuadas.

Se puede consultar en cualquier momento cuál fue la última tecla pulsada por el usuario con Tec_Ultima(), lo que resulta muy cómodo para obtener información en funciones muy alejadas de aquellas en las que el programa interactúa con el usuario.

Se puede pedir una tecla que pertenezca a un determinado rango con Tec_Validada() e incluso hacerlo con vuelta inmediata, si no hay tecla correcta disponible, con Tec_ValidadaRapido().

Hay funciones para pasar a mayúsculas o minúsculas, pero en esta versión sólo para caracteres ASCII (hasta 127).

La lista completa de macros está en Tecla.h. No se han definido todos los que son posibles en cada sistema operativo, sino sólo aquellos que son perfectamente compatibles.

Cuadro

Se pueden representar cuadros, cajas y líneas de cualquier color disponible y en dos grosores distintos.

El color se indica con Cdr_Color() y para representar los grosores de las líneas se tienen los macros CDR_SIMPLE y CDR_DOBLE.

Usuario

Las dos últimas filas de la pantalla están reservadas para interactuar con el usuario. Hay funciones para darle indicaciones, informarle de acciones y también para que edite enteros y texto (cadenas).

Fichero

En esta versión de Olimpo sólo se tienen funciones para manejar ficheros de texto. Se pueden abrir, cerrar y borrar. También leer y escribir líneas enteras de texto en ellos.

Las funciones se encargan de gestionar los errores que se puedan producir. Por ejemplo, al llamar a Fch_Borra(), primero se pide confirmación al usuario, luego

se comprueba la existencia del fichero y por último se emite un mensaje indicando si ha sido posible el borrado o no.

Menú

Hay menús horizontales y verticales. Sólo hay que preocuparse de asignar la zona de la pantalla en la que se desea que aparezca el menú, el objeto se ocupa de la colocación. Las opciones pueden llevar una tecla caliente, un atajo o "hotkey", que permite elegir la opción simplemente pulsando esa tecla. Para indicar cuál es, se utiliza el carácter '>' antes de la letra en la llamada al objeto.

Programa

Este objeto se ocupa simplemente de presentar información sobre el programa en la línea superior de la pantalla. Si no se utiliza, se dispone de una línea más para el desarrollo del programa.

Configuración

Este objeto permite leer información de un fichero filtrando todos los comentarios que se hayan escrito en él. El programador debe decidir después cómo lee la información entregada por el objeto.

En los ficheros de configuración se considera línea de comentario toda aquella que tenga un '*' (asterisco) como primer carácter, aunque haya caracteres en blanco antes de él.

También se pueden poner comentarios dentro de una línea poniendo "//" (doble barra). Todo lo que haya a continuación, se considera comentario.

Relación alfabética de funciones

```
FUNCION:  Abs()
OBJETIVO: Calcular el valor absoluto de un número
ENTRADAS: Un número
SALIDAS:  El valor absoluto
EJEMPLO:  Abs(-3)

FUNCION:  Azr_Entero ()
OBJETIVO: Obtener un número entero aleatorio en un rango especificado
ENTRADAS: a: número más bajo admitido; b: el más alto admitido
SALIDAS:  un número entero entre a y b (ambos inclusive)
EJEMPLOS: Azr_Entero (0,100)
          Azr_Entero (-10,10)

FUNCION:  Azr_Inicia ()
OBJETIVO: Iniciar el generador con un número aleatorio
ENTRADAS: Ninguna, se lee el reloj del sistema
SALIDAS:  Ninguna
EJEMPLO:  Azr_Inicia ()

FUNCION:  Cad_Cambia()
OBJETIVO: Cambiar en una cadena un carácter por otro
ENTRADAS: Cadena destino, carácter viejo, carácter nuevo
SALIDAS:  La cadena queda modificada y se devuelve
EJEMPLO:  Cad_Cambia (Cadena,'a','b')

FUNCION:  Cad_CaracterEnCadena (Caracter, Cadena)
OBJETIVO: Averiguar si el Caracter pertenece a la Cadena
ENTRADAS: El caracter y la cadena
SALIDA:   La posición que ocupa el carácter en la cadena, o 0
NOTA:     Se empieza a contar en 1
EJEMPLO:  Cad_CaracterEnCadena ('a',Nombre)

FUNCION:  Cad_ConvierteEntero()
OBJETIVO: Convertir un número entero en una cadena
ENTRADAS: La cadena y el número
SALIDAS:  La cadena queda modificada y se devuelve
EJEMPLO:  Cad_ConvierteEntero (Auxiliar,130)
```

```
FUNCION:  Cad_Copia()
OBJETIVO: Copiar una cadena en otra
ENTRADAS: La cadena destino y la cadena origen
SALIDAS:  La cadena destino
EJEMPLO:  Cad_Copia (Nombre,"Eustaquio")

FUNCION:  Cad_Longitud()
OBJETIVO: Calcular la longitud de una cadena
ENTRADAS: Una cadena
SALIDAS:  Su longitud
EJEMPLO:  Cad_Longitud (Nombre)

FUNCION:  Cad_Mueve()
OBJETIVO: Mover hacia la derecha todos los caracteres de una cadena
          desde una posición determinada, cuidando que la longitud
          no exceda un cierto máximo
ENTRADAS: La cadena, la posición y el máximo
NOTA:     Las posiciones se cuentan a partir de cero
SALIDAS:  La cadena queda modificada y se devuelve
EJEMPLO:  Cad_Mueve (Cadena,2,40)

FUNCION:  Cad_PrimerUtil (Cadena)
OBJETIVO: Decir la posición del primer carácter de una cadena que no
          sea ni blanco, ni tabulador, ni retorno de carro ni nueva línea
ENTRADAS: La cadena
SALIDA:   La posición que ocupa el primer carácter útil, o 0
NOTA:     Se empieza a contar en 1
EJEMPLO:  Cad_PrimerUtil ( "   Ahora" )

FUNCION:  Cad_QuitaCaracter()
OBJETIVO: Quitar el caracter que ocupa una determinada posición
          en una cadena
ENTRADAS: La cadena y la posición que hay que suprimir
NOTA:     Las posiciones se cuentan a partir de cero
SALIDAS:  La cadena queda modificada y se devuelve
EJEMPLO:  Cad_QuitaCaracter (Cadena,2)

FUNCION:  Cad_Subcadena (Larga,Corta)
OBJETIVO: Averiguar si la cadena Corta está contenida en la Larga
ENTRADAS: Las dos cadenas
SALIDA:   La posición que ocupa el primer carácter de Larga donde
          comienza la primera aparición de Corta, o 0
NOTA:     Se empieza a contar en 1
EJEMPLO:  Cad_Subcadena ( "Calamares","mar" )

FUNCION:  Cad_Trozo()
OBJETIVO: Extraer una cadena de otra
ENTRADAS: Cadena destino, cadena origen y puntos de comienzo y fin del
          corte
SALIDAS:  La cadena destino queda modificada y se devuelve
NOTA:     El primer carácter lleva el número 1
EJEMPLO:  Cad_Trozo (Subcadena,Nombre,3,7)

FUNCION:  Cad_Une()
OBJETIVO: Unir una cadena con otra
ENTRADAS: La cadena destino y la cadena origen
SALIDAS:  La cadena destino queda modificada y se devuelve
EJEMPLO:  Cad_Une (Nombre," Ruiz")

FUNCION:  Cdr_Caja()
OBJETIVO: Dibujar una caja
ENTRADAS: Tipo de la caja y sus coordenadas
SALIDAS:  Ninguna
EJEMPLO:  Cdr_Caja (CDR_DOBLE,1,1,10,70)

FUNCION:  Cdr_Color()
OBJETIVO: Establecer el color con el que se dibujarán todas las figuras
ENTRADAS: El color
SALIDAS:  Ninguna
```

```
EJEMPLO:  Cdr_Color ( ROJO )

FUNCION:  Cdr_Dibuja()
OBJETIVO: Dibujar un cuadro
ENTRADAS: El tipo de línea que se dibuja en las líneas externas,
          el de las internas, el número de filas, el alto de cada
          una, el número de columnas, el ancho de cada una, la fila
          y columna de la esquina superior izquierda
SALIDAS:  Ninguna
EJEMPLO:  Cdr_Dibuja (CDR_DOBLE,CDR_SIMPLE,2,3,4,1,3,3)

FUNCION:  Cdr_Linea()
OBJETIVO: Dibujar una línea
ENTRADAS: El tipo de línea y las coordenadas
SALIDAS:  Ninguna
EJEMPLO:  Cdr_Linea (CDR_DOBLE,2,3,2,71)

FUNCION:  Cnf_Abre()
OBJETIVO: Abrir un fichero de configuración
ENTRADAS: El nombre del fichero
SALIDAS:  El fichero abierto o NULO si ha habido algún error
EJEMPLO:  Cnf_Abre ( "Datos.dat" )

FUNCION:  Cnf_Cierra ()
OBJETIVO: Cerrar un fichero de configuración
ENTRADAS: El fichero
SALIDAS:  NULO si todo va bien, EOF si hay error
EJEMPLO:  Cnf_Cierra ( FicheroConfig )

FUNCION:  Cnf_Lee()
OBJETIVO: Obtener la siguiente línea de un fichero de configuración
ENTRADAS: El fichero y una cadena
SALIDAS:  El fichero si no hay error, o NIL; la cadena se modifica
NOTA:     La cadena no contiene \n
EJEMPLO:  Cnf_Lee ( Entrada, Cadena )

FUNCION:  Fch_Abre()
OBJETIVO: Abrir un fichero
ENTRADAS: El nombre del fichero y el tipo
SALIDAS:  El fichero abierto o NULO si ha habido algún error
EJEMPLO:  Fch_Abre ( "Datos.dat", FCH_LECTURA )

FUNCION:  Fch_AbreGrabar()
OBJETIVO: Abrir un fichero para grabar en él línea a línea
ENTRADAS: El nombre del fichero
SALIDAS:  El fichero abierto o NULO si ha habido algún error o negativa
EJEMPLO:  Fch_AbreGrabar ( "Informe.txt" )

FUNCION:  Fch_AbreLeer()
OBJETIVO: Abrir un fichero para leerlo línea a línea
ENTRADAS: El nombre del fichero
SALIDAS:  El fichero abierto o NULO si ha habido algún error
EJEMPLO:  Fch_AbreLeer ( "Datos.dat" )

FUNCION:  Fch_Borra()
OBJETIVO: Borrar un fichero
ENTRADAS: El nombre del fichero
SALIDAS:  Ninguna
EJEMPLO:  Fch_Borra ( "Viejo.txt" )

FUNCION:  Fch_Cierra ()
OBJETIVO: Cerrar un fichero
ENTRADAS: El fichero
SALIDAS:  NULO si todo va bien, EOF si hay error
EJEMPLO:  Fch_Cierra ( Salida )

FUNCION:  Fch_EscribeLinea()
OBJETIVO: Escribir una línea en un fichero (es decir, se pone \n)
ENTRADAS: El fichero y una cadena
SALIDAS:  El fichero si no hay error, o NIL
```

```
EJEMPLO:  Fch_EscribeLinea ( Salida, Linea )

FUNCION:  Fch_Existe()
OBJETIVO: Informar si un fichero existe
ENTRADAS: El nombre del fichero
SALIDAS:  Lógica
EJEMPLO:  Fch_Existe ( "Datos.dat" )

FUNCION:  Fch_LeeLinea()
OBJETIVO: Obtener la siguiente línea de un fichero
ENTRADAS: El fichero y una cadena
SALIDAS:  El fichero si no hay error, o NIL; la cadena se modifica
NOTA:     La cadena no contiene \n
EJEMPLO:  Fch_LeeLinea ( Entrada, Cadena )

FUNCION:  Fch_Nombre()
OBJETIVO: Formar el nombre completo de un fichero uniendo el nombre,
          el separador y la extensión
ENTRADAS: La cadena donde se pone el nombre completo, el nombre y
          la extensión
SALIDAS:  La cadena con el nombre completo
EJEMPLO:  Fch_Nombre ( NombreFichero, "Datos", "txt" )

FUNCION:  Max()
OBJETIVO: Calcular el máximo de dos números
ENTRADAS: Los números
SALIDAS:  El mayor de ellos
EJEMPLO:  Max (2,5)

FUNCION:  Men_Horizontal()
OBJETIVO: Presentar al usuario un menú horizontal para que pueda
          elegir entre varias opciones, mediante las teclas del
          cursor o mediante un atajo (tecla caliente)
ENTRADAS: La fila en la que se muestra el menú, las columnas entre
          las que se muestra, un vector de cadenas, terminado en
          NIL, en el que se señalan los atajos y la opción que hay
          resaltar en primer lugar
SALIDAS:  Un número indicando la opción elegida, empezando a contar
          en 1, o 0 si no se elige ninguna
EJEMPLO:  Men_Horizontal (1,1,78,{">Fichero","A>yuda",NIL},2)

FUNCION:  Men_Vertical()
OBJETIVO: Presentar al usuario un menú vertical para que pueda
          elegir entre varias opciones, mediante las teclas del
          cursor o mediante un atajo (tecla caliente)
ENTRADAS: Las coordenadas de la esquina superior izquierda e inferior
          derecha de la zona asignada al menú, un vector de cadenas,
          terminado en NIL, en el que se señalan los atajos y la opción que
          hay que resaltar en primer lugar
SALIDAS:  Un número indicando la opción elegida, empezando a contar
          en 1, o 0 si no se elige ninguna
EJEMPLO:  Men_Vertical (1,10,8,20,{">Fichero","A>yuda",NIL},2)

FUNCION:  Min()
OBJETIVO: Calcular el mínimo de dos números
ENTRADAS: Los números
SALIDAS:  El menor de ellos
EJEMPLO:  Min (2,5)

FUNCION:  Pan_Borra()
OBJETIVO: Borrar una zona de la pantalla
ENTRADAS: El color y las coordenadas
SALIDAS:  Ninguna
EJEMPLO:  Pan_Borra ( ROJO,3,3,15,15 )

FUNCION:  Pan_BorraLinea()
OBJETIVO: Limpiar una línea determinada
ENTRADAS: La línea
SALIDAS:  Ninguna
EJEMPLO:  Pan_BorraLinea ( 0 )
```

```
FUNCION:  Pan_Caracter()
OBJETIVO: Escribir un carácter en la pantalla
ENTRADAS: El carácter
SALIDAS:  Ninguna
EJEMPLO:  Pan_Caracter ('A')

FUNCION:  Pan_Cierra ()
OBJETIVO: Dejar la pantalla preparada para volver al sistema operativo
ENTRADAS: Ninguna
SALIDAS:  Ninguna
EJEMPLO:  Pan_Cierra ()

FUNCION:  Pan_Color ()
OBJETIVO: Cambiar el color del papel y la tinta
ENTRADAS: El color del papel y de la tinta
SALIDAS:  Ninguna
EJEMPLO:  Pan_Color ( VERDE, NEGRO )

FUNCION:  Pan_Cursor()
OBJETIVO: Colocar el cursor en una posición determinada
ENTRADAS: La fila y la columna
SALIDAS:  Ninguna
EJEMPLO:  Pan_Cursor ( 0,0 )

FUNCION:  Pan_CursorVisible()
OBJETIVO: Poner y quitar el cursor
ENTRADAS: Lógica
SALIDAS:  Ninguna
EJEMPLO:  Pan_CursorVisible ( SI )

FUNCION:  Pan_Define ()
OBJETIVO: Definir la pantalla principal del programa
ENTRADAS: Dos enteros que no se usan
NOTA:     Se usa una pantalla de 24 filas (de 0 a 23) y
          80 columnas (de 0 a 79)
EJEMPLO:  Pan_Define ( 24,80 )

FUNCION:  Pan_Entero()
OBJETIVO: Escribir un número entero en la pantalla en un ancho determinado
ENTRADAS: El número y el ancho
SALIDAS:  Ninguna
EJEMPLO:  Pan_Entero ( 12345,6 )

FUNCION:  Pan_Limpia ()
OBJETIVO: Borrar la pantalla completa
ENTRADAS: Ninguna
SALIDAS:  Ninguna
EJEMPLO:  Pan_Limpia()

FUNCION:  Pan_Papel ()
OBJETIVO: Cambiar el color de la pantalla
ENTRADAS: El nuevo color
SALIDAS:  Ninguna
EJEMPLO:  Pan_Papel ( VERDE )

FUNCION:  Pan_PonCar()
OBJETIVO: Escribir un carácter en cierta posición
ENTRADAS: La fila, la columna y el carácter
SALIDAS:  Ninguna
EJEMPLO:  Pan_PonCar (3,5,'A')

FUNCION:  Pan_PonEntero()
OBJETIVO: Escribir un entero en cierta posición
ENTRADAS: La fila, la columna, el entero y el ancho disponible
SALIDAS:  Ninguna
EJEMPLO:  Pan_PonEntero (3,5,12345,6)

FUNCION:  Pan_PonTexto()
OBJETIVO: Escribir un texto en cierta posición
```

```
      ENTRADAS: La fila, la columna y el texto
      SALIDAS:  Ninguna
      EJEMPLO:  Pan_PonTexto (3,5,"Hola")

      FUNCION:  Pan_Resalta()
      OBJETIVO: Poner y quitar la situación de resaltado de caracteres
      ENTRADAS: Lógica
      SALIDAS:  Ninguna
      EJEMPLO:  Pan_Resalta ( SI )

      FUNCION:  Pan_Texto()
      OBJETIVO: Escribir un texto en la pantalla
      ENTRADAS: Una cadena
      SALIDAS:  Ninguna
      EJEMPLO:  Pan_Texto ("Hola, usuario")

      FUNCION:  Pan_Tinta ()
      OBJETIVO: Cambiar el color con el que se escribe en la pantalla
      ENTRADAS: El nuevo color
      SALIDAS:  Ninguna
      EJEMPLO:  Pan_Tinta ( ROJO )

      FUNCION:  Prg_Presenta()
      OBJETIVO: Presentar el programa
      ENTRADAS: Nombre, versión, autor y fecha
      SALIDAS:  Ninguna
      EJEMPLO:  Prg_Presenta ( "Programa","0.0","Pedro Reina","1993" )

      FUNCION:  Son_Apaga()
      OBJETIVO: Apagar el sonido
      ENTRADAS: Ninguna
      SALIDAS:  La variable Son_Encendido se modifica y se devuelve
      EJEMPLO:  Son_Apaga()

      FUNCION:  Son_Bien()
      OBJETIVO: Señalar brevemente que algo es correcto
      ENTRADAS: Ninguna
      SALIDAS:  El valor de Son_Encendido
      EJEMPLO:  Son_Bien()

      FUNCION:  Son_Cambia()
      OBJETIVO: Cambiar el estado del sonido (Encendido / Apagado)
      ENTRADAS: Ninguna
      SALIDAS:  La variable Son_Encendido se modifica y se devuelve
      EJEMPLO:  Son_Cambia()

      FUNCION:  Son_Enciende()
      OBJETIVO: Encender el sonido
      ENTRADAS: Ninguna
      SALIDAS:  La variable Son_Encendido se modifica y se devuelve
      EJEMPLO:  Son_Enciende()

      FUNCION:  Son_Error()
      OBJETIVO: Señalar un error
      ENTRADAS: Ninguna
      SALIDAS:  El valor de Son_Encendido
      EJEMPLO:  Son_Error()

      FUNCION:  Son_MalaTecla()
      OBJETIVO: Señalar que la tecla pulsada no se admite
      ENTRADAS: Ninguna
      SALIDAS:  El valor de Son_Encendido
      EJEMPLO:  Son_MalaTecla()

      FUNCION:  Tec_Disponible()
      OBJETIVO: Decir si está disponible alguna tecla
      ENTRADAS: Ninguna
      SALIDAS:  0 si no hay tecla disponible, 1 si la hay
      EJEMPLO:  Tec_Disponible()
```

```
FUNCION:  Tec_FijadoMayus()
OBJETIVO: Decir si está activado el sujeta-mayúsculas
ENTRADAS: Ninguna, se lee del sistema
SALIDAS:  Lógica
EJEMPLO:  Tec_FijadoMayus()

FUNCION:  Tec_Mayus()
OBJETIVO: Convertir una tecla en mayúscula
ENTRADAS: Una tecla
SALIDAS:  La tecla convertida en mayúscula
NOTA:     En esta versión no se tienen en cuenta caracteres acentuados
EJEMPLO:  Tec_Mayus ( TEC_A )

FUNCION:  Tec_Minus()
OBJETIVO: Convertir una tecla en minúscula
ENTRADAS: Una tecla
SALIDAS:  La tecla convertida en minúscula
NOTA:     En esta versión no se tienen en cuenta caracteres acentuados
EJEMPLO:  Tec_Minus ( TEC_MAY_A )

FUNCION:  Tec_Pertenece()
OBJETIVO: Decidir si una tecla pertenece a cierto rango
ENTRADAS: Una tecla y un vector con las teclas admitidas, terminado en NIL
SALIDAS:  La tecla si pertenece o NIL si no pertenece
EJEMPLO:  Tec_Pertenece ( TEC_F1, {TEC_ESC,NIL} )

FUNCION:  Tec_Pulsada()
OBJETIVO: Esperar a que el usuario pulse una tecla y devolver
          su código
ENTRADAS: Ninguna
SALIDAS:  El código asignado a la tecla pulsada
          La variable global Tec_UltimaPulsada queda modificada
NOTA:     Deben usarse siempre los macros, puesto que los códigos
          dependen del compilador
EJEMPLO:  Tec_Pulsada()

FUNCION:  Tec_Ultima()
OBJETIVO: Devolver la última tecla pulsada por el usuario
ENTRADAS: Ninguna, se usa la variable global Tec_UltimaPulsada
SALIDAS:  El código asignado a Tec_UltimaPulsada
EJEMPLO:  Tec_Ultima()

FUNCION:  Tec_Validada()
OBJETIVO: Devolver una tecla de un determinado rango
ENTRADAS: Un vector con las teclas admitidas, terminado en NIL
SALIDAS:  La tecla pulsada
EJEMPLO:  Tec_Validada ( {TEC_ESC,NIL} )

FUNCION:  Tec_ValidadaRapido()
OBJETIVO: Devolver una tecla de un determinado rango o NIL si no hay
          disponible ninguna tecla correcta
ENTRADAS: Un vector con las teclas admitidas, terminado en NIL
SALIDAS:  La tecla pulsada o NIL
EJEMPLO:  Tec_ValidadaRapido ( {TEC_ESC,NIL} )

FUNCION:  Tim_Crono()
OBJETIVO: Obtener una referencia temporal de precisión
ENTRADAS: Ninguna, se consulta el reloj del sistema
SALIDAS:  Un número real indicando el número de segundos desde la
          medianoche

FUNCION:  Tim_Espera()
OBJETIVO: Suspender la ejecución del programa durante cierto tiempo
ENTRADAS: El tiempo en segundos
SALIDAS:  Ninguna

FUNCION:  Usr_Avisa()
OBJETIVO: Avisar al usuario de algo
ENTRADAS: El texto del aviso
SALIDAS:  Ninguna
```

```
EJEMPLO:  Usr_Avisa ( "Operación incorrecta" )

FUNCION:  Usr_BorraZona()
OBJETIVO: Borrar la zona de interacción con el usuario
ENTRADAS: El nuevo color de la zona
SALIDAS:  Ninguna
EJEMPLO:  Usr_BorraZona ( NEGRO )

FUNCION:  Usr_Consulta()
OBJETIVO: Preguntar algo al usuario
ENTRADAS: El texto de la pregunta
SALIDAS:  Lógica, según la contestación del usuario
EJEMPLO:  Usr_Consulta ( "Borrar este fichero" )

FUNCION:  Usr_Entero()
OBJETIVO: Devolver un número determinado por el usuario después de
          editar el que se le ofrece
ENTRADAS: El número que hay que editar, el ancho asignado, los valores
          mínimo y máximo admitidos, las coordenadas donde se edita y
          los colores
SALIDAS:  El entero editado
EJEMPLO:  Usr_Entero (100,4,-900,2000,1,1,BLANCO,NEGRO)

FUNCION:  Usr_Indica()
OBJETIVO: Mandar un mensaje al usuario
ENTRADAS: Dos cadenas
SALIDAS:  Ninguna
EJEMPLO:  Usr_Indica ( "Elige una opción", "Pulsa ENTER" )

FUNCION:  Usr_Informa()
OBJETIVO: Mandar un mensaje al usuario
ENTRADAS: El texto del mensaje
SALIDAS:  Ninguna
EJEMPLO:  Usr_Informa ( "Calculando" )

FUNCION:  Usr_PulsaUnaTecla()
OBJETIVO: Mandar un mensaje al usuario y esperar que pulse una tecla
ENTRADAS: El texto del aviso
SALIDAS:  Ninguna
EJEMPLO:  Usr_PulsaUnaTecla ( "Operación terminada." )

FUNCION:  Usr_Texto()
OBJETIVO: Devolver una cadena determinada por el usuario después de
          editar la que se le ofrece
ENTRADAS: La cadena que hay que editar, el ancho asignado,
          las coordenadas donde se edita y los colores
SALIDAS:  El entero editado
EJEMPLO:  Usr_Texto ("Artesonado",20,1,1,BLANCO,NEGRO)
```

EL PC VISTO DESDE EL QL

El PC que había antes del nacimiento del QL era muy lento en velocidad comparado con un QL original. La única ventaja que tenia sobre el QL eran sus unidades de disco de 5.25" (frente a los MDVs naturalmente). Ya en los días del nacimiento del QL y el MAC nacía el PC AT, que tenía una velocidad similar al MAC de 1984, y su mayor ventaja era su disco duro de 20 megas. Cuando murió oficialmente el QL, aún era más rápido que la mayoría de los PCs XT de aquel entonces (incluso el compilador TURBO de D.P. generaba un código más rápido que el generado por el TURBO-Basic en un 80386 a 8 MHz). Luego, con el paso de los años, el PC ha seguido evolucionando, mientras que el QL sigue siendo el mismo de 1984. Pero gracias a un milagro, en 1991 la firma MIRACLE SYSTEM lanza al mercado una tarjeta llamada GOLD CARD que soluciona la mayoría de los problemas de edad que tenía el viejo QL, la velocidad en especial (un 68000 a 16 MHz o 24 MHz es tan rápido como la mayoria de los PCs AT más rápidos).

La mayor ventaja del QL sobre el PC siempre ha sido su sistema operativo QDOS multitarea, y a pesar de que no han salido nuevas versiones del QDOS, y sí del MS-DOS, el QDOS de 1984 sigue siendo un sistema operativo a años luz del MS-DOS

de 1992.

Dentro de los sistemas operativos compatibles con QDOS, el único que he podido probar ha sido el MINERVA. Os puedo garantizar que MINERVA es lo que podría haber sido la siguiente versión del sistema operativo QDOS. En el mundo PC o MAC es seguro que sonaría a imposible un sistema operativo multitarea e intérprete BASIC multitarea en solamente una EPROM de 64 Kbytes (MINERVA). En el PC, para poder hacer lo mismo que QDOS (48K) o MINERVA (64K), el OS/2 v2.0 necesita la friolera de 21 discos 3.5" HD (¡¡Casi 30 megas!!).

Gracias al emulador PC CONQUEROR he podido ir estudiando el PC, pero lo único que he sacado bastante claro es que un PC no es la máquina que a mí me gustaria usar. Pero si no fuese usuario de QL o no hubiese conocido el QL, entonces sería usuario de PC, ya que no habría encontrado nada mejor que pudiese pagar dentro del rango de mi hispánico bolsillo. A lo largo de los años puedo decir que conozco muy bien el QL, y me oculta muy pocos secretos, pero también puedo garantizar que conozco y domino mejor un PC que la mayoría de los usuarios de PC. Para mi gusto, el mejor lenguaje para programar un PC y sacar el máximo de rendimiento es el 'C'. Con él podemos hacer prácticamente todo lo que puede dar el PC y su MS-DOS sin esfuerzo.

Si alguien está interesado en comprar un PC, que tenga cuidado con lo que compra. Hoy en día parece que a las casas de software para PC les ha dado por engordar los programas con pienso de engorde para que solamente puedan correr en PC AT con 1 Mb de RAM para arriba, un mínimo de 40 Mb de disco duro, y naturalmente una CPU rápida. Por ejemplo: TURBO C v1.0 se podía usar en un PC con dos unidades de 5.25" de 360 Kb. Ahora el TURBO C++ v1.01 necesita memoria extendida (el programa ocupa cerca de 900 Kb + los ficheros de configuración y una ayuda de unos 600 Kb) y su configuración normal ocupa 6 Mb de disco duro. Los mejores programas de contabilidad del mercado español necesitan memoria extendida y un disco duro mínimo de 20 Mb para guardar la aplicación (datos y programa), y todo gracias a que están escritos en DBASE III y compilados.

El TURBO C++ v1.01 he conseguido correrlo con PC CONQUEROR GOLD. En su configuración mínima se ha tragado casi todo el disco duro de 3 Mb. He tenido que prescindir de la versión independiente (uso la integrada) y de la mayoría de sus utilidades independientes. Para poder editar y compilar, no me ha quedado más remedio que utilizar un disco HD en la unidad A:. Comparando el TURBO C con el TURBO C++, aún no he acabado de comprender por qué ocupa tantos bytes, pues lo único nuevo que hay a primera vista es una nueva presentación en pantalla, pero no justifica el engorde de bytes del programa. A mí me da la impresión de que existe una mano negra detrás de todo esto que tiene como objetivo renovar el parque de compatibles PC (muy pronto los programas de PC dejarán de ser compatibles con el IBM PC XT).

Advertencia: PC CONQUEROR GOLD corriendo a 24 MHz da error de track 0 inválido y no sé qué al intentar formatear un disco DD o HD. En cambio a 16 MHz formatea estupendamente discos DD y HD.

Hoy en día, un QL con GOLD CARD, MINERVA, HERMES, discos ED y caja/teclado SPEM está en muy buena posición frente a los actuales PCs AT. La mayor ventaja de un QL frente a un PC sigue siendo su sistema operativo y su intérprete SuperBASIC. La única desventaja que tiene la configuración QL expuesta arriba frente a los actuales PC AT es su precio (Un PC AT con un Mb de RAM, VGA monocromo y disco de 20 Mb solamente cuesta 34.000 ptas).

Salvador Merino, Fuengirola, 30/9/92

PREOCUPACIONES SOBRE EL FUTURO DE QLIPER
==

Está finalizando el año 1992 (año 12 después de Sinclair). El QL ha conseguido sobrevivir nueve largos años gracias al esfuerzo e interés de sus usuarios, y un puñado de firmas pequeñas de hardware y software. Su situación actual es muy buena, y podría resistir por tiempo indefinido.

A pesar de que nuestro olvidado QL tiene posibilidades de un futuro prometedor con el nuevo SuperQL de Miracle, la tarjeta QL para PC y la tarjeta gráfica para QL, lamentablemente, estoy convencido de que nuestro Club QLíper va a perder un

minimo de 4 socios el próximo año 1993.

¿Que cómo estoy muy seguro? LLevo metido en esto desde hace seis años, y es tiempo suficiente para poder haber obtenido experiencia sobre el comportamiento de un usuario de ordenador. Las pruebas que hago para obtener resultados son secreto profesional, y no deseo poner nervioso al personal, pero os advierto que la mayoría de los socios no leen QLíper en el momento que lo reciben (Parece que cuesta trabajo pasarlo por la impresora, e incluso hay pocas ganas de verlo en pantalla).

La librería de software de dominio público no ha sido apenas usada durante 1992. Lo peor es que el socio que más ha usado la librería, solamente ha pedido la mitad, y más de la mitad de los socios nunca han pedido un disco.

En 1993 seremos menos socios que este año, pero aún podría darse el milagro de obtener nuevos socios (¡Nunca debemos perder la esperanza!). QLíper se puede mantener igual con 10 socios que con 100. La única diferencia es que cuantos más socios somos, más trabajo para el Editor. El Club se puede mantener fácilmente con 10 socios que se comprometan a escribir un mínimo de 2 páginas cada dos meses (¡¡dos líneas por día!!), lo que haría un total de cerca de 100 Kb. El resto del disco se puede llenar con pantallas digitizadas (no he enviado nuevas, porque nadie ha dicho nada ni me ha enviado nada para digitalizar), y programas PD de buena calidad que nadie ha solicitado (inéditos).

Actualmente estoy intentando convencer a varias personas del mundo de la CONTABILIDAD para que me ayuden en la escritura de un programa de contabilidad de calidad SUPER-comercial. Ese programa será escrito en 'C', y espero poder publicar una copia de la versión QL gratis para todos los socios de 1993.

A pesar de todas mis preocupaciones, 1993 será un buen año para el QL. Si mal no recuerdo, es el año en el cual AMSTRAD pierde los derechos del QL. Y si todo sale bien, es el año del nacimiento del super esperado SuperQL.

Salvador Merino, Fuengirola, 30/9/1992

CUADRICULANDO A 85 X 25

Cuando usaba el Spectrum vendían papel cuadriculado en formato de la pantalla del Spectrum. Con él se podían diseñar fondos de pantallas, e incluso diseñar nuestros gráficos. Hoy, con el proyecto de escribir un programa de contabilidad de calidad aceptable en un mundo tan competitivo como el MS-DOS, para poder diseñar bonitas pantallas necesitaba papel cuadriculado a 80 x 25 para fondos de pantallas en modo MDA. El papel cuadriculado que había en las librerías no me convencía para mis necesidades, así que me puse manos a la obra y a preparar con mi QL e impresora EPSON LQ-550 mi propio papel cuadriculado. Dado este plan, nació el programita en 'C' que podéis encontrar en el disco.

Dicho programa utiliza los caracteres gráficos ampliados de EPSON (e IBM). Cuadricula a 85 x 25, la pantalla QL en modo 4. Como ya podréis imaginar, la primera versión del programita fue escrita en TURBO C++ para PC. Teclear los caracteres gráficos me fue muy fácil, pues tengo configurado el driver KEY_DATA de PC CONQUEROR para trabajar con caracteres españoles y los caracteres gráficos más corrientes. El programa "cuadriculado_exe" es la versión compilada. Antes el diseño de una pantalla lo hacía improvisando, pero utilizando estas cuadrículas a la larga se tarda menos tiempo y queda muchísimo mejor.

Salvador Merino, Fuengirola, 1/10/1992

PREPARANDO LAS FUNCIONES QPTR EN C

Como ya sabéis por mis anteriores cartas, estaba trabajando en el proyecto de escribir un buen programa de CONTABILIDAD muy actualizado y fácil de usar. A pesar de que lo tenía todo y no veía ningún obstaculo, el proyecto lo veo cada día más lejos de convertirse en una realidad. ¿Por qué? Yo tengo mucha experiencia en programación, pero sobre contabilidad solamente conozco la contabilidad personal de negocios como venta al por menor, alquiler de coches,

hostal, APARHOTEL, taxis, bares, construcción y venta de viviendas... (advierto que para llevar ese tipo de contabilidad frefiero utilizar libros y carpetas), y desconozco totalmente el Plan General Contable Español. Los individuos que se suponía que iban a participar en el proyecto como asesores, están interesados en el proyecto, pero no tienen prisa. Aún no hemos escrito ni una sola línea del análisis funcional (luego viene el análisis órganico, la programación, los manuales,....), y lo único que me han enseñado es a jugar al MUS. Así que ya me veo jugando en el campeonato de MUS de la peña del CIMONIO (EL CIPOTE COMO UN DEMONIO).

Como veis, mis amigos de la infancia solamente usan el ordenador en el trabajo, y por necesidad. No están interesados en programar ni en leer manuales (los programas deben de ser muy fáciles de usar). Fuera del trabajo solamente están interesados en jugar y divertirse. Esto es muy normal en Andalucía, pero como solamente trabajo con europeos, se me olvida que yo también soy andaluz a pesar de que mis costumbres han cambiado tanto que ya ni me parezco a un español normal.

Me he desviado mucho del titulo. La presente carta es para advertiros de que no me he quedado de brazos cruzados, me he entretenido en probar las funciones QPTR para C68 que Tony Tebby a puesto en PD (Este paquete lleva ya muchos meses en nuestra librería y nadie lo ha pedido). El primer defecto que encontré fue que mis intentos por compilar los ejemplos con el C68 v3.01 fueron desastrosos. Después me entretuve en leer detenidamente los manuales, y lo único que encontré bastante claro fue que las funciones habían sido escritas con el compilador GST para el C68 v1.04. Más tarde, observando todos los ficheros, me di cuenta de que los errores inexplicables estaban en los ejemplos
.
Con mucha paciencia fui depurando cada ejemplo hasta que se conseguía su correcta compilación y ejecución.

El QPTR de Qjump nació en 1986 como una parte integral del sistema operativo del QLT o FUTURA de SANDY. En 1987 Tony Tebby decidió poner a la venta el primer programa que usaba el QPTR, el QRAM (Serafín Olcoz consiguió el QRAM mucho antes de su comercialización como copia beta, y yo lo obtuve como regalo al comprar mi TRUMP CARD en Mayo'87). Luego más tarde Qjump puso en venta el QPTR con un Toolkit para el SuperBASIC. Pasaron los años y nadie hizo uso del QPTR en España. Nacho Enrique Cabero fue el único que dijo que estaba probando las maravillosas posibilidades sin explotar del QPTR, pero se compró un Commodore AMIGA 2000 antes de terminar su compilador COBOL para QL, sistema operativo MINIX para QL, y su trabajo sobre QPTR. Yo en cambio me quedé esperando ejemplos y comentarios sobre el QPTR en QUANTA o QL WORLD. Lo único que he podido saber es que programar bajo el interface QPTR desde SuperBASIC era bastante complicado (algunos usuarios escribieron rutinas que facilitaban las cosas), y era muchísimo más sencillo hacerlo desde ensamblador. A partir de 1988 aparecieron en el continente los primeros programas que utilizaban el interface QPTR. Hoy la mayoría de los programas continentales comerciales utilizan el interface QPTR.

Si el QL vuelve al mundo comercial, el interface usuario QPTR de Qjump (Tony Tebby) es nuestra principal arma frente a los interfaces usuario de otros ordenadores. Lo bueno del caso es que el entorno QPAC II no tiene competencia. Utilizar el interface QPTR desde 'C' es muchísimo más sencillo y cómodo que hacerlo desde ensamblador (y de SuperBASIC mejor no hablar). Me temo que la mayoría de los futuros programas QL serán escritos en 'C' en un futuro no muy lejano.

Para utilizar los programas compilados con C68 v3.01 que utilicen la librería QPTR se necesita una copia del QPTR y Window Manager (podéis conseguirla en QRAM, QPAC I, QPAC II, QPTR, y programas comerciales que utilicen el QPTR).

Salvador Merino, Fuengirola, 26/10/92

EL RINCON DEL PINCHA-DISCOS

¿Sabéis cuantos discos me habéis pedido desde la bajada de precios? ¡NINGUNO! Esto ya se convierte en preocupante, pues la caída de los pedidos va unida a la caída de los precios de los compatibles PC, lo que significa que muchos socios se han comprado un PC, y no leen la revista en disco QLíper (leer poco es muy

normal en el español medio, tanto que los PCs AT más baratos se entregan sin manuales para abaratar costes). Menos mal que con la llegada de la tarjeta QL para PC de MIRACLE, el panorama podría cambiar.

Viendo que nadie pide programas por las buenas, tomo la drástica solución de utilizar la fuerza. En otras palabras, enviar programas muy seleccionados para su publicación en la revista QLíper. Los primeros programas son:

- Versión operativa light del QPTR para C68.- A diferencia del disco que tenemos en la librería, todo funciona bien con el C68 v3.01. Estoy preparando un pequeño cursillo para aquellos usuarios que no posean la documentación QPTR. También es muy probable que escriba una nueva versión de los programas 'C' utilizando el interface QPTR.

- XTRICATOR v1.10 (la más reciente versión).- La versión comentada era la v1.00. Este magnífico emulador ZX 81 no ha sido pedido por nadie. Aquellos que poseen la v1.00 ha sido porque se la he copiado sin preguntar. Hay dos discos llenos de programas ZX 81 comerciales en su mayoría, pero solamente se pueden copiar con un copiador sectorial, o utilizando el QZdriver.

A mí me da lo mismo copiar los discos por 50 ptas que enviarlos poco a poco para su publicación en QLíper, pero utilizando la segunda opción podemos tardar una eternidad (calculando por lo bajo, unos 30-40 años).

Salvador Merino, Fuengirola, 3/11/1992

YA EXISTE EL EMULADOR DE QL PARA ATARI MEGA STE y TT

QVME es el nombre del emulador de QL para ATARI STE y TT. Soporta resoluciones de 1024x768, 896x896...

Aunque sé muy poco sobre el QVME, tiene muchas posibilidades de tratarse de una tarjeta para BUS VME, lo que significa que con solamente modificar su software podria utilizarse en otros ordenadores con BUS VME (P.e.: Algunos AMIGAs y MACs).

Salvador Merino, 3/11/1992

POWER-PC-601, EL BEBE NACIDO DEL ACUERDO IBM, APPLE Y MOTOROLA

La producción conjunta del nuevo microprocesador de última generación discurre más rápidamente de lo previsto. El primer prototipo ha sido presentado a finales de octubre del 92.

El PowerPc-601 es un microprocesador tipo RISC, que es una tecnología que limita el número de comandos a ejecutar para incrementar la velocidad de transmisión de las informaciones. Este microprocesador incluye 2,8 millones de transistores (el primer 68000 tenía unos 68000 transistores) en una pastilla de cuatro décimas de pulgada de lado y el bus 88110 de Motorola, lo que permite aplicar el producto a una amplia gama de sistemas, incluyendo ordenadores personales, estaciones de trabajo y sistemas multiprocesador.

Estos nuevos chips serán utilizados por la firma Apple en las próximas versiones de sus ordenadores Macintosh para conseguir una mayor eficacia en el procesamiento de la información. Sin embargo, estos computadores conservarán sus características empleando el mismo lógico.

Las tres firmas tienen previsto una nueva alianza para el próximo año bajo el nombre de "PowerOpen Association".

El modelo 601 es el primero de una familia de procesadores POWER PC, que incluirá cuatro modelos diferentes y que podría significar un paso importante en la ruptura del monopolio de los chips con arquitectura Intel 80x86 en el campo de los ordenadores personales.

Según mi opinión, es muy agradable ver cómo la familia 680XX va a ver prolongada su existencia con el nuevo microprocesador RISC PowerPc-601. También debo recordar que la familia 680XX tiene garantizada su existencia en una posible

nueva generación de ordenadores de 64 bits, pues su juego de instrucciones fue creado pensando en esa posibilidad.

Salvador Merino, Fuengirola, 5/11/1992

RECIBI EL DIPLOMA DE ANALISTA PROGRAMADOR EMITIDO POR CEAC
==

Hoy he recibido el diploma, que certifica haber demostrado mi capacidad en los estudios del curso de ANALISTA PROGRAMADOR, previo pago de un último sablazo de 2.000 ptas.

El diploma es una cartulina de tamaño doble folio, que mucho me temo que no posee ningún valor, pues hay un texto que dice:

"Estos estudios no tienen efectos académicos oficiales y están comprendidos en el ámbito de aplicación del R. Decreto 2641/1980 de 7 Noviembre (B.O.E. del 12 diciembre. Por lo que se expide el presente diploma. En Barcelona, a 2 de septiembre de 1992."

Como podéis observar, el diploma como recuerdo es bonito (casi igual que el Graduado Escolar y el Título de Bachiller), pero no sirve para nada.

El contenido el curso de Analista Programador era bastante interesante, sencillo de aprender y muy aplicable a una situación real. Su único inconveniente es su precio, 116.800 ptas por 2.000 páginas de texto y 200 folios de exámenes me sigue pareciendo muy caro. Eso sí, si tuviese que ir a estudiar a la Universidad en Málaga, solamente el primer año de curso me habría gastado mucho más en transporte (tren).

Si tenemos en cuenta que mi objetivo era dominar mejor lo que ya sabía, creo que ha sido alcanzado.

En lo referente a una posible incursión comercial, lo he estado pensando mejor, y echando números. Ahora que la tarjeta QL para PC va a nacer el mismo día que el niño Dios (Navidad), estoy más interesado por el sistema ShareWare, pues lo veo ideal para no pagar impuestos y no tener que darme de alta en nada raro que me pueda acarrear más problemas con Hacienda y compañía.

ShareWare es el nombre que dan los anglosajones a un sistema de venta muy peculiar. El programador distribuye su programa gratis por mediación de BBS y librerías de software de dominio público. El usuario que obtiene el programa, puede hacer dos cosas: utilizarlo gratis, o registrarse (enviar una cuota para obtener la última versión y tener derecho a futuras actualizaciones, o simplemente poder consultar o advertir BUGs). Este sistema de venta es de locos, pero tiene sus ventajas: no pagar impuestos, ahorro en marketing...

Salvador Merino, 6/11/1992

XTRICATOR, EL EMULADOR DE ZX 81 PARA QL
=======================================

El ZX 81 es el segundo ordenador comercializado por Tío Clive Sinclair. En 1980 (nuestro año cero después de Sinclair), introdujo su primer ordenador, el ZX 80, el cual fue un éxito con 50.000 unidades vendidas (tenía 4 Kbytes de ROM, 1 Kbyte de RAM y solamente trabajaba con enteros).

El ZX 81 venía con 8 Kbytes de ROM, 1 Kbyte de RAM, ofrecía operaciones en coma flotante, una pantalla de 32x24 en modo texto y 64x48 en modo gráfico en blanco y negro. El procesador Z80A trabajaba a 3.2 MHz. La primera versión costaba 70 libras (por aquel entonces un IBM MS-DOS PC con 64 Kbytes y disco de 180 Kbytes costaba 3.000 libras). En los primeros 3 meses de existencia del ZX 81 se vendieron 50.000 unidades, y durante 1981-83 la cifra fue de 4.5 millones de unidades.

XTRICATOR es un emulador Z80 que emula un Sinclair ZX 81 en un Sinclair QL y sus clonos. Es compatible con todas las versiones de la ROM QL (incluida Minerva). También es compatible con el QPTR de Qjump.

Una vez cargado Xtricator en nuestro QL con GOLD CARD más que un ZX 81 parece un Super-ZX 81 por las siguientes razones:

- La velocidad es el doble con GOLD CARD (30% en QL 68008 a 7.5 MHz).
- La ROM contiene los 8 Kbytes de ROM de un ZX 81 estándar + CORAL BASIC INTERPRETER en la zona 8-16 K de ROM. Este nuevo BASIC se arranca con RAND USR 8192. Tiene todos comandos ZX 81 + 46 nuevos (incluye procedimientos, estructuras lazo e intercambio de datos en ambos sentidos), los cuales son introducidos letra por letra (no por pulsación de una sola tecla).
- Hi-Res de 256x192 en blanco y negro.
- La ZX PRINTER es emulada en SER1.
- QZ, un manejador de ficheros en disco y en disco RAM.
- Sin contar que bajo QDOS y QPTR de QJUMP podemos tener varios ZX 81 trabajando en multitarea.

Una vez cargado el emulador lo primero que hay que hacer es consultar el teclado de un ZX 81. Esto se hace con CTRL+F1, y se retorna con F1 (para no consultar tanto, más valdría hacerse con una foto del teclado del ZX 81 o memorizar las teclas más usadas).

Pasar los programas ZX 81 a QL es una operación que no está al alcance de cualquiera, porque se necesita un ZX 81 con interface serie RS-232 (el cual se puede construir por solamente 5 libras). Claro que en el supuesto de tener un ZX 81 que no se use desde hace más de 8 años, podéis haceros una idea en el estado de conservación que deben estar las cintas de cassette con programas.

Para aquellos que deseen programas ZX 81, os puedo garantizar que no será un problema. Carlo Delhez (autor del emulador ZX 81) ha preparado una colección en disco de QL con 200 programas del ZX 81 (la librería QLíper dispone de 80 programas para ZX 81, la mayoría, o todos, comerciales en su época). Otros medios para obtener programas para ZX 81 es aprovecharse de aquellos disponibles en disco ST o MS-DOS para los emuladores de ZX 81 para ATARI ST e IBM PC.

Carlo Delhez sueña con el éxito de Xtricator y quizás revivir las antiguas glorias del ZX 81. Lo cierto es que la aventura que he tenido con el XTRICATOR me ha servido para revivir viejos tiempos que para muchos parecen la prehistoria de la microinformática y lo ven como algo muy lejano, porque no la han vivido, pero para mí, es como si hubiese sido ayer. Tenía 17 años por aquel entonces, me faltó el canto de un duro para comprar un ZX 81, pues si el ZX SPECTRUM se hubiese retrasado tan sólo un mes en su aparición en España, mi segundo ordenador habría sido un ZX 81.

Probar los programas del ZX 81 puede dar la sensación de habernos introducido en el túnel del tiempo. Esto nos da la oportunidad de ver cómo eran los programas de hace 11 años a una velocidad doble de la tenían en su día.

El disco 1 de la librería ZX 81 contiene 80 programas. Entre ellos podéis encontrar:

-FIG-FORTH, para los amantes del lenguaje Forth.
-Ensamblador, desensamblador y editor, para aquellos dispuestos a seguir escribiendo programas en código máquina del Z80.
-MCODER-2, un compilador de BASIC.
-FLIGHT, el simulador de vuelo de Psion, prácticamente igual que el del Spectrum, pero más sencillo.
-LA PULGA, primer programa de Paco Suárez comercializado por la firma Indescomp. Fue un éxito, pero la versión para Spectrum lo fue más por sus mejoras en gráficos.
-INVADERS, el famoso "Space Invaders" de siempre. Se puede jugar a varias velocidades, pero en un QL con Gold Card hasta la velocidad más lenta podría resultar demasiado rápida.
-KRZYKONG.- El famoso juego de King Kong lanzando barriles.
-SCRAMBL81 y SCRAMBLE, dos versiones del famoso juego en el que con nuestra nave tenemos que ir recorriendo varias pantallas, bombadeando y disparado a todo misil y fuel, o más fácil, a todo lo que se ponga por delante.
-GALAXIANS, como su nombre indica, el famoso juego donde los marcianos abandonan la formación para atacarnos.
-GUNFIGHT, un juego para dos o uno solo; dos pistoleros en duelo y solamente una carreta entre ambos.

-BREAKOUT, el clásico rompedor de muros.
-FROGGER, el famoso juego de la rana que cruza la autopista y el río para llegar a casa. Esta versión tiene muy buenos gráficos, y es incluso demasiado rápida en velocidad 0 (0-4). Este programa usa DAA, y para correrlo con Xtricator hay que hacer lo siguiente:

```
LOAD "FROGGER"
-Pulsar ESC y CTRL-
POKE 17250,0
POKE 21725,0
RUN
```

No voy a seguir nombrando, pues no he podido tener paciencia para verlos todos, pero hasta llegar a los ochenta hay una base de datos, un "toolkit" gráfico, programas educativos y un montón de juegos.

Todo usuario de Sinclair QL debería tener XTRICATOR, más que nada porque se trata de nuestro abuelo dentro de la familia de ordenadores Sinclair, el ZX 81.

Gracias a Carlo Delhez, hoy tenemos a toda la familia reunida dentro de un único ordenador Sinclair (ZX 81, ZX Spectrum y QL).

Aprovecho para recordar que hasta la caída de Sinlcair en 1986, los ordenadores Sinclair dominaban el mundo y no había marca que le hiciese sombra en unidades vendidas. A pesar de los años transcurridos desde entonces, prácticamente ninguna marca de ordenadores ha conseguido batir al ZX 81 o al ZX Spectrum en unidades vendidas (incluso el Atari ST y el Amiga necesitaron varios años para alcanzar al QL). Y si el QL no hubiese sido eliminado del mapa en 1986, mucho dudaría yo de que los PCs dominasen hoy el mundo.

Salvador Merino, 4/5/1992

SPECTATOR, EL EMULADOR DE ZX SPECTRUM + INTERFACE I

Actualmente tengo la versión 0.62, pero el autor, Carlo Delhez, ha prometido enviarme la v1.00 en el otoño del 92, pues soy usuario registrado. Poco hay que contar sobre este emulador que ya no hayáis observado, pues teóricamente, deberiais estar ya leyendo el manual y probándolo, porque la v0.62 se va a publicar en QLíper.

La velocidad en un QL con 68000 a 16 MHz es aproximadamente el 30% de la velocidad de un ZX Spectrum. Los usuarios de QL estándar mejor abtenerse, pues solamente vais a obtener resultados parecidos a los que daba el emulador de PC "The Solution" en su día.

Para obtener programas ZX Spectrum solamente hay tres soluciones:

1- Conectar un QL y un Spectrum vía RS-232.
2- Utilizar las utilidades conversoras de disco.
3- Aprovecharse de los programas para Spectrum ya disponibles en disco para emuladores de Spectrum para PC, ST y QL.

En el supuesto de que tengamos un Spectrum con Interface I, el cable RS-232 QL-ZX, y un interfaz copiador como el Trans-Tape (Todo eso, si mal no recuerdo, lo tengo yo almacenado en un armario), muy laborioso veo yo el tinglado que hay que liar de cables, ladrones, monitores, ordenadores, etc... Para mi gusto, lo más preferible es que un usuario de Spectrum te pase los programas en disco 3.5" o 5.25" y luego le demos una pasada al disco con las utilidades conversoras (es lo más fino, limpio y sencillo).

Como en nuestra librería QLíper tenemos dos discos llenos de programas para Spectrum, he podido probar unos pocos (he salido mareado de tanto juego) para poder contar mi experiencia con Gold Card:

COMMANDO: Se puede jugar bastante bien. Buen "scroll". Solamente he conseguido llegar hasta final segunda fase. En mis mejores tiempos llegaba más lejos.
ANT ATTACK: El famoso juego de las hormigas en tres dimensiones. En mis mejores tiempos, rescataba a la chica con los ojos cerrados (hoy, mejor no hablar). La

velocidad es tan buena como en el Spectrum.
SABRE WULF: Nuestro explorador se mueve muy rápido. Se trata de una joya que no ha perdido su atractivo original.
MANIC MINER: El original juego de plataformas. Sigue siendo rápido y adictivo.
FRED: Uno de los primeros programas españoles, si no es el primero, que triunfaron en el mercado británico e internacional. El problema principal es el "scroll" del programa, que al redibujar toda la pantalla produce un pestañeo, pero sigue siendo un programa histórico.
ALIEN 8: Segundo programa de Ultimate que usaba la técnica "filmation". El robot se pasea por los habitaculos de la nave en tres dimensiones. El programa debe de ser muy rápido, porque con el emulador sigue siendo rápido.
RAIDERS: Una de invasores espaciales bastante jugable.
HUNGRY HORACE: Uno de los primeros juegos de Psion en la saga Horace. Es tan rápido como en el Spectrum.
HIGHWAY: El famoso robot y su columna de robots que intentan arrastrar un objeto a través de una autopista llena de peligros. Muy jugable y adictivo.
COMBAT: El mítico juego del helicóptero de guerra con más teclas de lo normal. Si alguien tiene interés en jugarlo, sigue siendo rápido. Lo que rogaría es que os tomarais en broma el mensaje de la recompensa si enviais los datos de quien os ha proporcionado la copia ilegal, porque mi copia procede del mundo PC.

En resumen, quitando un ensamblador de Z80 llamado Thornado y una utilidad, todo son juegos. A pesar de que el sonido en el Spectrum había que escucharlo con trompetilla, lo echo un poco de menos. La pantalla del Spectrum en el monitor QL es más grande, pero parece "cinemascope". Probablemente os ponga un juego de Spectrum junto a este comentario para que vayáis probando el emulador. Para cargarlo : LOAD "/SABRE" Cuando reciba la v1.00 os contaré qué tiene de nuevo. Se supone que será más compatible todavía con el Spectrum.

Salvador Merino, 21/5/1992

LOS EMULADORES DE ZX SPECTRUM
=============================

Actualmente existen 3 emuladores Spectrum para QL, Spectator (Holanda), ZX (RUSIA) y ZX EMU (Italia). Aunque si tenemos en cuenta que el segundo consta de 3-4 versiones muy diferentes, deberíamos hablar de 5-6 emuladores de ZX SPECTRUM.

El emulador que yo poseo es el SPECTATOR v 0.62 (el mismo que envié a QLíper). El autor tiene que enviarme la v1.00 cuando la tenga terminada, pero sé que ya ha salido la v0.96, y me tiene advertido que puedo actualizarlo sin coste adicional, pero voy a esperar la v1.00 que debe de estar al salir de un momento a otro.

El SPECTATOR v0.62 no puede correr todo el software Spectrum, pero sí una buena cantidad de programas (Hay dos discos llenos en la libreria QLíper). La v1.00 podría correr prácticamente casi todo el software Spectrum, pero el autor dice que será un poco más lenta que las versiones previas. La velocidad de la versión 0.62 es muy irregular, pero a 24 MHz es muy interesante. Juegos como el Manic Miner, el Sabre, Alien 8, Raiders, Zombies... corren un poco más rápido que en un Spectrum REAL, pero otros como Commando son un poquito más lentos (muy poco). Para mi gusto, a 24 MHz se puede jugar muy bien con los viejos juegos del Spectrum.

EL emulador ZX EMU se puede obtener en tres versiones. Una de ellas es el ZM EMU-1, el cual es ShareWare (se ha publicado en la revista QITALY, y se puede obtener de muchas librerías PD. P.e: de QLIPER cuando tengamos pasta para hacer pedidos de nuevos discos PD), y hay otra nueva vesión ShareWare llamada ZX EMU-PLUS que podría haber salido ya. Las siguientes dos versiones son ya el llamado ZM-x system que es un programa comercial que comercializa ERGON DEVELOPMENT (Italia). Consta de ZM-1+ (versión ShareWare), ZM-2 (versión emulador comercial), y ZM-3 (Compilador ZX).

ZM-x System es un programa que no poseo aún, y no puedo opinar ni hacer pruebas para probar si son ciertas las velocidades que ponen en su propaganda (80%-120% en GOLD CARD a 24 MHz). Una diferencia grande es que puede emular el sonido del Spectrum, y el modo SUPERVISOR es muy apetitoso. Otra diferencia es la emulación del sistema de almacenamiento de cassette en disco.

Están trabajando en el ZM-4, el cual podría alcanzar la velocidad de un Spectrum REAL en una GOLD CARD a 16 MHz.

ZM-x System lo comercializa: Davide Santachiara, Via Emilio De Marchi 2, IT-42100 Reggio Emilia, ITALIA. Cuesta 77000 Liras + 11000 Liras de embalaje y envío.

El emulador ZX de Andrew Lavrov (Rusia) es totalmente desconocido para mí. Lo único que sé es que Andrew Lavrov es el diseñador del sistema operativo para el ordenador ruso CONCEPT (un compatible QL basado en el MC 68040 a 40 MHz).

Hace muchos años, cuando nació la fiebre de los emuladores en 1986-7, el problema que tenían aquellos primeros emuladores era la velocidad. Hoy el QL, gracias a la maravillosa GOLD CARD, puede emular viejas máquinas a la misma velocidad, o incluso triplicar la velocidad primitiva original en algunos casos (P.e.: ZX-81 y CP/M). Lo que significa que hoy da gusto trabajar con un emulador. Antes había que darle un poco de paciencia a la cosa, y no ponerse histérico si aquello iba más lento de lo que podíamos soportar.

Gracias a los emuladores de SPECTRUM, el software del Spectrum no se perderá totalmente, y las generaciones futuras podrán estudiar u observar cómo eran las máquinas que hicieron llegar la informática a nuestras casas. Hoy el Spectrum ha caído después de 10 largos años, ya no es aquel ordenador de superventas, pero ha ganado la inmortalidad (algo que otros ordenadores, que compitieron con él, no han conseguido).

Salvador Merino, 13/11/1992

SPECTATOR V1.03

Ya me ha enviado Carlo Delhez la versión oficial del programa Spectator. La v1.03 es muy compatible con el ZX Spectrum. Todos los programas disponibles en disco QL corren estupendamente. ¿Qué tiene de nuevo Spectator v1.03 no disponible en versiones anteriores? Lo primero que llama la atención es la emulación del sonido. Hay dos modos:

- Utilizando el altavoz del QL.- No produce el sonido real de un Spectrum, pero algo es algo.
- Utilizando la QL NET.- El sonido producido es muy similar al producido por un Spectrum REAL. Se necesitan unos auriculares o un altavoz conectado al puerto NET del QL.

El menú principal del Spectator tiene las siguientes opciones:

- Volcado de la pantalla del QL.
- Volcado de la pantalla del Spectrum (Screen$).
- Ayuda sobre el teclado.
- Información extra.
- Ayuda sobre las principales teclas de función.
- Finalizar Spectator.
- Resetear el Z80.
- Salvar el contenido de la memoria en un fichero comprimido Z80.

Además, tiene nuevas utilidades para conversión de programas de otros formatos de disco.

En el tema de velocidad, tengo que decir que para algunos programas prefiero utilizar la vieja v0.62, pues el SABRE a 24 MHz corre más rápido que un Spectrum REAL en la v0.62 y en la v1.03 se hunde en velocidad el muñeco (los bichos no). En cambio otros programas como el COMMANDO, que era lento en la v0.62, ahora es tan rápido como en un Spectrum REAL en la v1.03.

Tengo pensado pedir la versión comercial del ZM-x System a Italia el próximo año, pero voy a esperar hasta mediados de enero, pues el autor de Spectator ya dice que está preparando la v1.30. No ha especificado detalles, pero la nueva versión podría leer cintas de cassette Spectrum sin la ayuda de un Spectrum (¿Cómo? Z88 lo hace desde el interface serie RS-232). Aunque lo que sí va a

tener con seguridad es opciones para cambiar los discos por defecto, y la posibilidad de usar un poderoso desensamblador de Z80, un monitor de memoria y una rutina para trazar el código y examinar los juegos y utilidades.

Lo único que me podría atraer del ZM-x system es su velocidad en modo compilación, pero en ese modo la compatibilidad es bastante baja. Si hay socios interesados en compartir conmigo el ZM-x system a lo mejor decido comprarlo.

Salvador Merino, 21/11/1992

"INTELIGENCIA ARTIFICIAL" EN ARCHIVE

El presente artículo se dedica a describir el funcionamiento del programa "GENINFOR", que he desarrollado en los últimos meses. No os envío copia del programa por distintas razones: en primer lugar su utilización es estrictamente especializada y dirigida a un sector muy limitado (genera informes de exploración psicológica), por otra parte requiere la utilización de todo mi sistema de gestión integrada para el gabinete. En todo caso, si entre vosotros existe (cosa que dudo) algún psicólogo en activo interesado en el sistema puede ponerse en contacto conmigo en la dirección o el teléfono de siempre.

Como ya he mencionado el programa GENINFOR objeto de este comentario se inserta en un sistema mucho más amplio para la gestión integrada de un gabinete psicológico. Dicho sistema consta de tres grupos de funciones principales, además de algunas funciones secundarias. Esos tres apartados principales se encargan respsctivamente de gestionar: 1, el fichero de pacientes en Reeducación; 2, el fichero de pacientes en Tratamiento de Conducta y 3, la Facturación y la Contabilidad.

GENINFOR trabaja sobre los ficheros del área de Reeducación. De cara al usuario, este área se presenta como un fichero en el que la ficha correspondiente a cada paciente consta de seis páginas diferentes que puede consultar o modificar cómodamente mediante el uso de las teclas de cursor izquierda y derecha, mientras que el paso de un paciente a otro se realiza a traves de las flechas arriba y abajo. Internamente la estructura es algo más compleja: se trata de un sistema de seis ficheros enlazados entre sí por un campo de código en común que se asigna automáticamente a cada paciente durante la creación de la ficha, de tal forma que el fichero principal (Datos Administrativos) se ordena en cada caso según el criterio de acceso, Apallidos, Fecha, Orden de creación, Psicólogo encargado del caso, Diagnóstico; mientras que los restantes cinco ficheros, que contienen los datos de la exploración realizada al niño (Factores generales, Gnosias, Praxias, Lecto-Escritura y Conducta), se encuentran siempre ordenados por el campo Código, de manera que una vez encontrada la ficha de Datos administrativos de un paciente según el criterio elegido, el programa se encarga de seleccionar las cinco fichas que con el mismo código se encuentran en los cinco ficheros restantes. La presntación al usuario se realiza a través de seis pantallas previamente diseñadas y almacenadas en disco en forma de ficheros _scr, que se cargan cuando se pide la correspondiente página de la ficha.

El programa GENINFOR se basa en los datos contenidos en los seis ficheros descritos para redactar un informe completo de la exploración. Dicho informe no se limita a la típica presentación de informes mecanizados que ofrecen algunas casas de mecanización de informes para colegios y que suelen reducirse a unos gráficos de asteriscos o a la simple enumeración de factores con su resultados numéricos. Cuando decidí crear GENINFOR me propuse obtener el mismo tipo de informe que hasta el momento nos encargábamos nosotros de redactar mediante Quill, que huye de los datos numéricos mientras no sean estrictamente necesarios y se dedica a una explicación del significado de los resultados lo más comprensible que se pueda de cara a las familias, aún manteniendo el rigor técnico y científico requerido en un informe de este tipo. Todo ello me conducía a la necesidad de que GENINFOR emitiese unos informes más cercanos al ser humano que a la máquina. Otra dificultad en el planteamiento del programa consiste en que debe describir y analizar más de 30 factores con un total de 107 campos de información relevante, que, para mayor inri, no están definidos según criterios unitarios, sencillos de analizar según reglas únicas, sino que se construyeron en su momento de acuerdo con la estructura de los datos tal como se obtienen de las pruebas psicológicas aplicadas y para una mayor facilidad de comprensión por parte del usuario psicólogo.

Para enfrentar estas dificultades el programa GENINFOR dispone de un

fichero principal, cuyos registros constan de los siguientes campos:

- NUM$, campo alfanumérico en el que se asigna un número de orden a cada campo de los ficheros del sistema.

- TITULO$, campo alfanumérico que incluye un título descriptivo del campo en cuestión, que resulta de utilidad para la redacción y el propio manejo del fichero.

- VALOR$, campo alfanumérico que contiene cada uno de los valores que puede tomar el campo original. Cuando los valores originales son numéricos, se encuentran transformados en alfanuméricos mediante la función str().

- PARRAFO$, campo alfanumérico que contiene la o las frases que utilizará GENINFOR para la descripción de los resultados obtenidos en el factor analizado.

- VALOR, es un campo numérico mediante el cual se unifica la escala de valoración de los distintos campos para su posterior análisis conjunto.

Para cada paciente, GENINFOR crea un fichero _dbf, que contiene el informe bajo el siguiente formato:

. un registro para cada frase y por cada registro:
   - un campo de preámbulo con códigos de control de impresora
   - un campo de texto con el contenido de la frase
   - un campo de postámbulo con códigos de control de impresora

Al principio del programa es posible seleccionar una de entre dos opciones: enviar el informe a impresora y al fichero simultáneamente o sólo al fichero.

El informe se genera de la siguiente manera:

- Membrete, con nombre y dirección del gabinete
- Datos administrativos, seleccionados de entre los que componen la ficha principal del paciente.
- Motivo de consulta y anamnesis, se detallan las causas de la consulta y un resumen de la historia clínica y evolutiva previas del paciente. Los datos se extraen del fichero de conducta y anamnesis.
- Pruebas aplicadas, donde simplemente se enumeran las pruebas utilizadas en la exploración.
- Resultados obtenidos donde se describen e interpretan los resultados obtenidos por pruebas y factores explorados:
   . Inteligencia: se analizan 11 puntuaciones parciales y 3 cocientes de inteligencia diferentes, señalando su significación clínica.
   . Coordinación Viso-Motora: se describen los resultados de la prueba aplicada para este factor.
   . Gnosias: se van describiendo los resultados obtenidos en los apartados Visuales, Auditivas, Tactiles, Dígito-Manuales, Temporales, Espaciales, Esquema Corporal y Conocimiento Derecha-Izquierda.
   . Praxias: se van describiendo los resultados obtenidos en los apartados Dígito-Manuales, Motricidad Facial, Movimientos Simultáneos y Alternos, Equilibrio, Ritmo, Sincinesias y Lateralidad.
   . Lectura: se describen los resultados en cuanto a comprensión, velocidad, entonación, repeticiones y silabeo. Así mismo se detalla la existencia o no de errores de tipo disléxico (Omisiones de palabra, renglón o letra, añadidos, sustituciones, transposiciones, inversiones, reordenamientos o lectura en espejo).
   . Escritura: se describen los resultados en cuanto a calidad grica, trazado de caracteres circulares, posición del lápiz en la mano y presión sobre el papel. Así mismo se detalla la existencia o no de errores de tipo disléxico (Omisiones, añadidos, uniones indebidas, separaciones indebidas, sustituciones, transposiciones, inversiones, reordenamientos o escritura en espejo).
   . Conducta: se describe el comportamiento del paciente según cinco factores diferentes, a partir de un cuestionario rellenado por los padres, así como el comportamiento del paciente durante la exploración.
- Coclusiones: a partir de una serie de índices que se van creando mediante el contenido del campo VALOR de cada factor analizado, se emiten unas conclusines sobre inteligencia, grado y equilibrio de desarrollo, posibles daños

orgánicos, nivel de aprendizaje en lecto-escritura y comportamiento.

- Recomendaciones: a partir de los mismos índices, se extraen las recomendaciones oportunas sobre la necesidad de seguir o no distintos tipos de tratamiento y sobre la necesidad o no de realizar algunas exploraciones complementarias (Oftalmológica, auditiva o neurológica).
- Fecha
- Nombre y número de colegiado del Psicólogo que realizó la exploración.

Como se puede apreciar, parte de la dificultad del proyecto residía en la enorme cantidad de apartados del informe, sin embargo esto no es ni con mucho el aspecto principal. De hecho la dificultad de la parte descriptiva del informe, que es la más voluminosa, reside en la necesidad de obtener una redacción en una prosa razonablemente legible. El aspecto más interesante desde el punto de vista informático se concentra en algunos puntos concretos del informe, a saber: Inteligencia, Conclusiones y Recomendaciones. En estos tres puntos el programa debe tomar decisiones a partir del análisis de datos múltiples y realizar por su cuenta la parte más "humana" del trabajo que realiza un psicólogo al emitir un informe; de ahí que me haya atrevido a hablar de "inteligencia artificial" y que me pueda referir a GENINFOR como un pequeño "sistema experto".

El resultado final es un informe sobre papel y sobre soporte magnético, de entre 4 y 5 páginas, realizado por el lentísimo Archive en algo menos de 10 minutos, mientras que con anterioridad suponía unas 5 horas de trabajo ante el teclado de Quill y una copia escrita de las fichas del paciente.

Como todo padre, guardo junto con el orgullo de haber engendrado a GENINFOR una cierta inseguridad en su capacidades, por lo que el programa presenta en pantalla y permite editar el texto en algunos puntos delicados, como son Motivo de Consulta y Anamnesis, Pruebas Aplicadas, Conclusiones y Recomendaciones.

Indepentinetemente de la utilidad clarísima que supone el ahorro de tiempo y esfurzo, GENINFOR ha supuesto para mí un nuevo reto para profundizar en las auténticas posibilidades de Archive, que una vez más no me ha defraudado.

José Carlos de Prada

ULTIMAS NOTICIAS

Precio de la tarjeta QXL con 8 Mb: 495 libras.
Precio de la versión reducida (memoria por determinar): 295 libras.
La fotografía de la tarjeta aparecida en SQLW es de tamaño real. Esto quiere decir que al parecer puede instalarse en un PC portátil.

Salvador Merino, abril de 1993

En este disco de QLíper van incluidos unos ficheros dbf de Archive con el contenido de los años 1990, 1991 y 1992 de la revista Sinclair QL Word. Provienen del disco 11 de la librería del club holandés SIN_QL_AIR, que se ha recibido aquí junto con el último número de su revista impresa (ayudadme los que gustéis de leer sólo literatura original y por tanto dominéis el holandés como vuestra propia lengua materna, entre otras decenas de cientos, claro). Creo que os será útil poder selecionar por temas o autores lo aparecido en SQLW.

Marcos, abril de 1993

LAS CUENTAS DE QLIPER

Fecha	Concepto	Ptas.	Saldo
93.01.01	Saldo anterior		+6820
93.01.11	Suscripción de Javier ZUBIETA	+2000	+8820
93.01.12	Suscripción de Miguel ESTARELLAS	+2000	+10820
93.01.12	De Miguel Estarellas por envío anterior	+200	+11020
93.01.14	Suscripción de Josu REGIDOR	+2000	+13020
93.01.14	Suscripción de Francisco DIAZ-TENDERO	+2000	+15020
93.01.14	Suscripción de Pablo CARDENAS	+2000	+17020
93.01.14	De Francisco y Pablo por envío anterior	+500	+17520

93.01.14	Suscripción de Diego MORIARTY	+2000	+19520
93.01.14	Comisión robada por el Banco Santander	-300	+19220
93.02.04	11 sobres acolchados	-396	+18824
93.02.04	Suscripción de Mariano BERGES	+2000	+20824
93.02.04	Sellos para QLíper 42	-2006	+18818
93.02.22	Sellos para QLíper 42 a SIN_QL_AIR	-135	+18683
93.02.22	Suscripción de Pedro REINA	+2000	+20683
93.02.22	Contribución de Pedro REINA	+1000	+21863
93.03.01	Suscripción de José Carlos DE PRADA	+2000	+23863
93.03.02	Suscipción de Felipe BERGANZA	+2000	+25863
93.03.23	20 discos Verbatim	-1790	+24073
93.04	25 sobres acolchados	-900	+23173

Total actual en caja: .... +23173

Los discos de QLíper:

Fecha	Cantidad	Total
========	===============	=====
91.10.23	+50	+50
92.01.07	-20 (QLíper 36)	+30
92.03.03	+50	+80
92.03.04	-20 (QLíper 37)	+60
92.04.25	-20 (QLíper 38)	+40
92.06.30	-20 (QLíper 39)	+20
92.08.24	+50	+70
92.09.09	-34 (QLíper 40)	+36
92.11.17	+50	+86
92.11.18	-34 (QLíper 41)	+52
93.02.04	-27 (QLíper 42)	+25
93.02.22	-1 (QLíper 42)	+24
93.03.24	+20	+44
	-05 (Redacción)	+39
		=====

Discos disponibles para enviar: .. +39